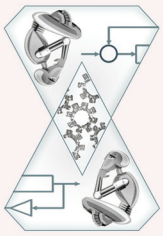


ICINCO 2009

6TH INTERNATIONAL CONFERENCE ON INFORMATICS IN CONTROL, AUTOMATION AND ROBOTICS

Proceedings

Volume 1 - Intelligent Control Systems and Optimization

MILAN - ITALY · JULY 2 - 5, 2009

INSTICC PRESS

ORGANIZED BY

INSTICC

IN COOPERATION WITH

aaai

CO-SPONSORED BY

IFAC

ICINCO 2009

Proceedings of the
6th International Conference on
Informatics in Control, Automation and Robotics

Volume 1
Intelligent Control Systems and Optimization

Milan, Italy

July 2 - 5, 2009

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

Co-Sponsored by
IFAC – International Federation of Automatic Control

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

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

Printed in Portugal
ISBN: 978-989-8111-99-9
Depósito Legal: 294961/09

http://www.icinco.org
secretariat@icinco.org

BRIEF CONTENTS

INVITED SPEAKERS

Daniel S. Yeung

University of Technology

China

Maria Pia Fanti

Polytechnic of Bari

Italy

Janan Zaytoon

University of Reims Champagne Ardennes

France

Alessandro Giua

University of Cagliari

Italy

Peter S. Sapaty

Institute of Mathematical Machines and Systems, National Academy of Sciences

Ukraine

ORGANIZING AND STEERING COMMITTEES

CONFERENCE CHAIR

Joaquim Filipe, Polytechnic Institute of Setúbal / INSTICC, Portugal

PROGRAM CHAIR

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

Jean-Louis Ferrier, University of Angers, France

PROCEEDINGS PRODUCTION

Sérgio Brissos, INSTICC, Portugal

Helder Coelhas, INSTICC, Portugal

Vera Coelho, INSTICC, Portugal

Andreia Costa, INSTICC, Portugal

Bruno Encarnação, INSTICC, Portugal

Bárbara Lima, INSTICC, Portugal

Raquel Martins, INSTICC, Portugal

Carla Mota, INSTICC, Portugal

Vitor Pedrosa, INSTICC, Portugal

José Varela, INSTICC, Portugal

CD-ROM PRODUCTION

Elton Mendes, INSTICC, Portugal

Pedro Varela, INSTICC, Portugal

GRAPHICS PRODUCTION AND WEBDESIGNER

Marina Carvalho, INSTICC, Portugal

SECRETARIAT AND WEBMASTER

Marina Carvalho, INSTICC, Portugal

PROGRAM COMMITTEE

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

Alessandro Chiuso, Universita di Padova, Italy

Hyo-Sung Ahn, Gwangju Institute of Science and Technology (GIST), Korea, Republic of

Frank Allgower, University of Stuttgart, Germany

Francesco Amigoni, Politecnico di Milano, Italy

Plamen Angelov, Lancaster University, U.K.

Peter Arato, Budapest University of Technology and Economics, Hungary

Helder Araújo, University of Coimbra, Portugal

Marco Antonio Arteaga, Universidad Nacional Autonoma de Mexico, Mexico

Vijanth Sagayan Asirvadam, Universiti Technologi PETRONAS, Malaysia

T. Asokan, Indian Institute of Technology Madras, India

Robert Babuska, Delft University of Technology, The Netherlands

Ruth Bars, Budapest University of Technology and Economics, Hungary

Adil Baykasoglu, University of Gaziantep, Turkey

Maren Bennewitz, University of Freiburg, Germany

Karsten Berns, University of Kaiserslautern, Germany

Arijit Bhattacharya, Dublin City University, Ireland

Sergio Bittanti, Politecnico di Milano, Italy

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

Jean-louis Boimond, ISTIA - LISA, France

Djamel Bouchaffra, Oakland University, U.S.A.

Bernard Brogliato, INRIA, France

Edmund Burke, University of Nottingham, U.K.

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

Dídac Busquets, Universitat de Girona, Spain

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

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

Alessandro Casavola, University of Calabria, Italy

Riccardo Cassinis, University of Brescia, Italy

Chien Chern Cheah, Nanyang Technological University, Singapore

Tongwen Chen, University of Alberta, Canada

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

Graziano Chesi, University of Hong Kong, China

Carlos Coello Coello, Cinvestav-IPN, Mexico

Yechiel Crispin, Embry-riddle Aeronautical University, U.S.A.

Michael A. Demetriou, Worcester Polytechnic Institute, U.S.A.

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

Jorge Dias, Institute of Systems and Robotics, Portugal

Gamini Dissanayake, University of Technology, Sydney, Australia

Denis Dochain, Université Catholique de Louvain, Belgium

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

Alexandre Dolgui, Ecole des Mines de Saint Etienne, France

Prabu Dorairaj, Wipro Technologies, India

Marco Dorigo, Université Libre de Bruxelles, Belgium

Venky Dubey, Bournemouth University, U.K.

Petr Ekel, Pontifical Catholic University of Minas Gerais, Brazil

Andries Engelbrecht, University of Pretoria, South Africa

Sebastian Engell, Univeristy of Dortmund, Germany

PROGRAM COMMITTEE (CONT.)

Simon G. Fabri, University of Malta, Malta

Sergej Fatikow, University of Oldenburg, Germany

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

Paolo Fiorini, Università degli Studi di Verona, Italy

Juan J. Flores, University of Michoacan, Mexico

Georg Frey, German Researc Center for Artificial Intelligence - DFKI, Germany

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

John Qiang Gan, University of Essex, U.K.

Nicholas Gans, National Reasearch Council and Air Force Research Laboratory, U.S.A.

Leonardo Garrido, Monterrey Institute of Technology, Mexico

Andrea Garulli, Universita' di Siena, Italy

Lazea Gheorghe, Technical University of Cluj-Napoca, Romania

Paulo Gil, Universidade Nova de Lisboa, Portugal

Alessandro Giua, University of Cagliari, Italy

Luis Gomes, Universidade Nova de Lisboa, Portugal

Dongbing Gu, University of Essex, U.K.

Guoxiang Gu, Louisiana State University, U.S.A.

Jason Gu, Dalhousie University, Canada

Wail Gueaieb, University of Ottawa, Canada

José J. Guerrero, Universidad de Zaragoza, Spain

Thomas Gustafsson, Luleå University of Technology, Sweden

Maki K. Habib, Saga University, Japan

Hani Hagras, University of Essex, U.K.

Wolfgang Halang, Fernuniversitaet, Germany

Riad Hammoud, Delphi Corporation, U.S.A.

Uwe D. Hanebeck, Institute for Anthropomatics, Germany

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

Dominik Henrich, University of Bayreuth, Germany

Francisco Herrera, University of Granada, Spain

Victor Hinostroza, University of Ciudad Juarez, Mexico

Wladyslaw Homenda, Warsaw University of Technology, Poland

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

Marc Van Hulle, K. U. Leuven, Belgium

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

Atsushi Imiya, IMIT Chiba University, Japan

Hisao Ishibuchi, Osaka Prefecture University, Japan

Thira Jearsiripongkul, Thammasat University, Thailand

Dimitrios Karras, Chalkis Institute of Technology, Greece

Dusko Katic, Mihailo Pupin Institute, Serbia

Graham Kendall, University of Nottingham, U.K.

Tamas Keviczky, Delft University of Technology, The Netherlands

Jonghwa Kim, University of Augsburg, Germany

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

Waree Kongprawechnon, Thammasat University, Thailand

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

George L. Kovács, Hungarian Academy of Sciences, Hungary

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

Kemal Leblebicio, Middle East Technical University, Turkey

Graham Leedham, University of New England, Australia

Kauko Leiviskä, University of Oulu, Finland

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

Tsai-Yen Li, National Chengchi University, Taiwan

PROGRAM COMMITTEE (CONT.)

Yangmin Li, University of Macau, China

Huei-Yung Lin, National Chung Cheng University, Taiwan

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

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

Jose Tenreiro Machado, Institute of Engineering of Porto, Portugal

Frederic Maire, Queensland University of Technology, Australia

Om Malik, University of Calgary, Canada

Jacek Mandziuk, Warsaw University of Technology, Poland

Hervé Marchand, INRIA, France

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

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

Carlo Menon, Simon Fraser University, Canada

Sanya Mitaim, Thammasat University, Thailand

Pieter Mosterman, The MathWorks, U.S.A.

Rafael Muñoz-salinas, University of Cordoba, Spain

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

Andreas Nearchou, University of Patras, Greece

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

Sergiu Nedevschi, Technical University of Cluj-Napoca, Romania

Anton Nijholt, University of Twente, The Netherlands

Hendrik Nijmeijer, Eindhoven University of Technology, The Netherlands

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

Urbano Nunes, University of Coimbra, Portugal

José Valente de Oliveira, Universidade do Algarve, Portugal

Romeo Ortega, LSS/CNRS/Supélec, France

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

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

Christos Panayiotou, University of Cyprus, Cyprus

Stefano Panzieri, Università degli Studi "Roma Tre", Italy

Igor Paromtchik, INRIA, France

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

Marie-Noëlle Pons, CNRS, France

Raul Marin Prades, Jaume I University, Spain

Jerzy Respondek, Silesian University of Technology, Poland

A. Fernando Ribeiro, Universidade do Minho, Portugal

Robert Richardson, University of Leeds, U.K.

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

Juha Röning, University of Oulu, Finland

António Ruano, CSI, Portugal

Fariba Sadri, Imperial College London, U.K.

Carlos Sagüés, University of Zaragoza, Spain

Mehmet Sahinkaya, University of Bath, U.K.

Antonio Sala, Universidad Politecnica de Valencia, Spain

Abdel-badeeh Salem, Ain Shams University, Egypt

Mitsuji Sampei, Tokyo Institute of Technology, Japan

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

Jurek Sasiadek, Carleton University, Canada

Daniel Sbarbaro, Universidad de Concepcion, Chile

Carla Seatzu, University of Cagliari, Italy

João Sequeira, Instituto Superior Técnico / Institute for Systems and Robotics, Portugal

PROGRAM COMMITTEE (CONT.)

Michael Short, University of Leicester, U.K.

Silvio Simani, University of Ferrara, Italy

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

Adam Slowik, Koszalin University of Technology, Poland

Michael Small, Hong Kong Polytechnic University, Hong Kong

Burkhard Stadlmann, University of Applied Sciences Wels, Austria

Tarasiewicz Stanislaw, Université Laval, Canada

Karl Stol, University of Auckland, New Zealand

Olaf Stursberg, Technische Universitaet Muenchen, Germany

Chun-Yi Su, Concordia University, Canada

Cornel Sultan, Virginia Tech, U.S.A.

Ryszard Tadeusiewicz, AGH University of Science and Technology, Poland

Choon Yik Tang, University of Oklahoma, U.S.A.

Daniel Thalmann, VR Lab EPFL, Switzerland

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

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

Nikos Tsourveloudis, Technical University of Crete, Greece

Anthony Tzes, University of Patras, Greece

Dariusz Ucinski, University of Zielona Gora, Poland

Nicolas Kemper Valverde, Universidad Nacional Autónoma de México, Mexico

Eloisa Vargiu, University of Cagliari, Italy

Laurent Vercouter, Ecole Nationale Superieure des Mines de Saint-Etienne, France

Bernardo Wagner, University of Hannover, Germany

Axel Walthelm, sepp.med GmbH, Germany

Dianhui Wang, La Trobe University, Australia

Qing-Guo Wang, National University of Singapore, Singapore

Zidong Wang, Brunel University, U.K.

James Whidborne, Cranfield University, U.K.

Dirk Wollherr, Technische Universität München, Germany

Marek Zaremba, Université du Québec (UQO), Canada

Janan Zaytoon, University of Reims Champagne Ardennes, France

Qin Zhang, University of Illinois at Urbana-Champaign, U.S.A.

AUXILIARY REVIEWERS

Andrea Baccara, Università degli Studi di Cagliari, Italy

Rui Cortesao, University of Coimbra, Portugal

Matteo de Felice, ENEA, Italy

Andrea Gasparri, University of Roma Tre, Italy

Zhi Han, The MathWorks, U.S.A.

Vitor Jorge, Universidade Federal do Rio Grande do Sul, Brazil

Enrico di Lello, Universita' degli Studi "Roma Tre", Italy

Andrea Monastero, University of Verona, Italy

Federico di Palma, University of Verona, Italy

Maura Pasquotti, Universita' degli Studi di Verona, Italy

Katalin Popovici, The MathWorks, France

Monica Reggiani, University of Padua, Italy

Maurizio di Rocco, Università degli Studi Roma Tre, Italy

P. Lopes dos Santos Santos, Instituto de Sistemas e Robótica - Porto, Portugal

Thomas Tometzki, Process Dynamics and Operations Group, Germany

Ali Emre Turgut, IRIDIA, Belgium

Maja Varga, Faculty of Electrical Engineering and Computing, Croatia

Kai Wurm, Institute of Computer Science, Germany

SELECTED PAPERS BOOK

A number of selected papers presented at ICINCO 2009 will be published by Springer-Verlag in a LNEE Series book. This selection will be done by the Conference chair and Program Co-chairs, among the papers actually presented at the conference, based on a rigorous review by the ICINCO 2009 Program Committee members.

FOREWORD

This book contains the proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2009) which was organized by the Institute for Systems and Technologies of Information, Control and Communication (INSTICC) and held in Milan. ICINCO 2009 was co-sponsored by the International Federation for Automatic Control (IFAC) and held in cooperation with the Association for the Advancement of Artificial Intelligence (AAAI).

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

In the Conference Program we have included oral presentations (full papers and short papers) and posters, organized in three simultaneous tracks: "Intelligent Control Systems and Optimization", "Robotics and Automation" and "Systems Modeling, Signal Processing and Control". We have included in the program five plenary keynote lectures, given by internationally recognized researchers, namely - Daniel S. Yeung (University of Technology, China), Maria P. Fanti (Polytechnic of Bari, Italy), Janan Zaytoon (University of Reims Champagne Ardennes, France), Alessandro Giua (Universita' di Cagliari, Italy) and Peter S. Sapaty (Institute of Mathematical Machines and Systems, National Academy of Sciences Ukraine).

The meeting is complemented with three satellite workshops, focusing on specialized aspects of Informatics in Control, Automation and Robotics; namely, the International Workshop on Artificial Neural Networks and Intelligent Information Processing (ANNIIP), the International Workshop on Intelligent Vehicle Controls & Intelligent Transportation Systems (IVC & ITS) and the International Workshop on Networked Embedded and Control System Technologies: European and Russian R&D Cooperation (NESTER).

ICINCO received 365 paper submissions, not including those of workshops, from more than 55 countries, in all continents. To evaluate each submission, a double blind paper review was performed by the Program Committee. Finally, only 178 papers are published in these proceedings and presented at the conference. Of these, 129 papers were selected for oral presentation (34 full papers and 95 short papers) and 49 papers were selected for poster presentation. The full paper acceptance ratio was 9%, and the oral acceptance ratio (including full papers and short papers) was 35%. As in previous editions of the Conference, based on the reviewer's evaluations and the presentations, a short list of authors will be invited to submit extended versions of their papers for a book that will be published by Springer with the best papers of ICINCO 2009.

Conferences are also meeting places where collaboration projects can emerge from social contacts amongst the participants. Therefore, in order to promote the development of re-

search and professional networks the Conference includes in its social program a Conference and Workshops Social Event & Banquet in the evening of July 4 (Saturday).

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

Commitment to high quality standards is a major aspect of ICINCO that we will strive to maintain and reinforce next year, including the quality of the keynote lectures, of the workshops, of the papers, of the organization and other aspects of the conference. We look forward to seeing more results of R&D work in Informatics, Control, Automation and Robotics at ICINCO 2010.

Joaquim Filipe
Polytechnic Institute of Setúbal / INSTICC, Portugal

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

Jean-Louis Ferrier
University of Angers, France

CONTENTS

SHORT PAPERS

INVITED SPEAKERS

KEYNOTE SPEAKERS

SENSITIVITY BASED GENERALIZATION ERROR FOR SUPERVISED LEARNING PROBLEM WITH APPLICATIONS IN MODEL SELECTION AND FEATURE SELECTION

Daniel S. Yeung
University of Technology
China

Abstract: Generalization error model provides a theoretical support for a classifier's performance in terms of prediction accuracy. However, existing models give very loose error bounds. This explains why classification systems generally rely on experimental validation for their claims on prediction accuracy. In this talk we will revisit this problem and explore the idea of developing a new generalization error model based on the assumption that only prediction accuracy on unseen points in a neighbourhood of a training point will be considered, since it will be unreasonable to require a classifier to accurately predict unseen points "far away" from training samples. The new error model makes use of the concept of sensitivity measure for an ensemble of multiplayer feedforward neural networks (Multilayer Perceptrons or Radial Basis Function Neural Networks). Two important applications will be demonstrated, model selection and feature reduction for RBFNN classifiers. A number of experimental results using datasets such as the UCI, the 99 KDD Cup, and text categorization, will be presented.

BRIEF BIOGRAPHY

Daniel S. Yeung (Ph.D., M.Sc., M.B.A., M.S., M.A., B.A.) is the President of the IEEE Systems, Man and Cybernetics (SMC) Society, a Fellow of the IEEE and an IEEE Distinguished Lecturer. He received the Ph.D. degree in applied mathematics from Case Western Reserve University. In the past, he has worked as an Assistant Professor of Mathematics and Computer Science at Rochester Institute of Technology, as a Research Scientist in the General Electric Corporate Research Center, and as a System Integration Engineer at TRW, all in the United States. He was the chairman of the department of Computing, The Hong Kong Polytechnic University, Hong Kong, and a Chair Professor from 1999 to 2006. Currently he is a Chair Professor in the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. His current research interests include neural-network sensitivity analysis, data mining, Chinese computing, and fuzzy systems. He was the Chairman of IEEE Hong Kong Computer Chapter (91and 92), an associate editor for both IEEE Transactions on Neural Networks and IEEE Transactions on SMC (Part B), and for the International Journal on Wavelet and Multiresolution Processing. He has served as a member of the Board of Governors, Vice President for Technical Activities, and Vice President for Long Range Planning and Finance for the IEEE SMC Society. He co-founded and served as a General Co-Chair since 2002 for the International Conference on Machine Learning and Cybernetics held annually in China. He also serves as a General Co-Chair (Technical Program) of the 2006 International Conference on Pattern Recognition. He is also the founding Chairman of the IEEE SMC Hong Kong Chapter.

His past teaching and academic administrative positions include a Chair Professor and Head at Department of Computing, The Hong Kong Polytechnic University, the Head of the Management Information Unit at the Hong Kong Polytechnic University, Associate Head/Principal Lecturer at the Department of Computer Science, City Polytechnic of Hong Kong, a tenured Assistant Professor at the School of Computer Science and Technology and an Assistant Professor at the Department of Mathematics, both at Rochester Institute of Technology, Rochester, New York.

He also held industrial and business positions as a Technical Specialist/Application Software Group Leader at the Computer Consoles, Inc., Rochester, New York, an Information Resource Sub-manager/Staff Engineer at the Military and Avionics Division, TRW Inc., San Diego, California, and an Information Scientist of the Information System Operation Lab, General Electric Corporate Research and Development Centre, Schenectady, New York.

IMPACT OF THE ICT ON THE MANAGEMENT AND PERFORMANCE OF INTELLIGENT TRANSPORTATION SYSTEMS

Maria Pia Fanti
Department of Electrical and Electronic Engineering, Polytechnic of Bari, Via Re David 200, Bari, Italy
fanti@deemail.poliba.it

Keywords: Intelligent Transportation Systems, Urban Traffic Control, Intermodal Transportation Networks, Information and Communication Technologies.

Abstract: Intelligent Transportation Systems (ITS) modelling, planning, and control are research streams that, in the last years, have received a significant attention by the researcher and practitioner communities due not only to their economic impact, but also to the complexity of decisional, organizational, and management problems. Indeed, the increasing complexity of these systems and the availability of the modern ICT (Information and Communication Technologies) for the interaction among the different decision makers and for the acquisition of information by the decision makers, require both the development of suitable models and the solution of new decision problems. This presentation is aimed at showing the new attractive researches and projects in the field of ITS operational control and management in Europe. In particular, it points out the key solution of using effectively and efficiently the latest developments of ICT for ITS operational management.

1 INTRODUCTION

The term Intelligent Transportation Systems (ITS) is used to refer to technologies, infrastructure, and services, as well as the planning, operation, and control methods to be used for the transportation of person and freight. In particular, Information and Communication Technologies (ICT) are considered to be the key tools to improve efficiency and safety in transportation systems. Indeed, the advent of ICT has a tremendous impact on the planning and operations of freight transportation and on traffic management systems. ITS technologies increase the flow of available data, improve the timeliness and quality of information and offer the possibility to control and coordinate operations and traffic in real-time. Significant research efforts are required to adequately model the various planning and management problems under ITS and real-time information, and to develop efficient solution methods.

In recent years, the European Union has sponsored several projects targeting advancements of different transportation systems. On the other hand, ITS topics are considered relevant and attractive research areas.

In Section 2 the paper recalls the most important European Projects in the fields of ITS and intelligent freight transportation. Moreover, Sections 3 and 4 present the research advances in two crucial sectors of ITS: the management of Urban Traffic Networks (UTN) and of Intermodal Transportation Networks (ITN), respectively.

2 EUROPEAN PROJECTS IN THE FIELD OF ITS

A basic project on ITS is CESAR I & II (Co-operative European System for Advanced Information Redistribution) that proposes an Internet communication platform that aims to integrate services and data for unaccompanied traffic and the rolling motorway traffic management. Moreover, in the field of railway system management, CroBIT (Cross Border Information Technology) is a new system that provides the railways with a tool to track consignments and integrates freight railways along a transport corridor providing total shipment visibility. A maritime navigation information structure in

European waters is established by MarNIS (Maritime Navigation Information Systems) that is an integrated project aiming to develop tools that can be used to exchange maritime navigation information and to improve safety, security and efficiency of maritime traffic.

In addition, several projects focus specifically on efficient freight transportation. For instance, Freightwise aims to establish a framework for efficient co-modal freight transport on the Norwegian ARKTRANS system. One of the main objectives in Freightwise is establishing a framework for efficient co-modal freight transport and simplifying the interaction among stakeholders during planning, execution and completion of transport operations. Moreover, the project e-Freight is a continuation of Freightwise to promote efficient and simplified solutions in support of cooperation, interoperability and consistency in the European Transport System. E-Freight is to support the Freight Logistics Action Plan, which focuses on quality and efficiency for the movement of goods, as well as on ensuring that freight-related information travels easily among modes. Furthermore, in the Seventh Framework Program (FP7-ICT Objective 6.1), the SMARTFREIGHT project wants to make urban freight transport more efficient, environmentally friendly and safe by answering to challenges related to traffic management and the relative coordination. Indeed, freight distribution management in city centres is usually operated by several commercial companies and there is no coordination of these activities in a way that would benefit the city. The main aim of SMARTFREIGHT is therefore to specify, implement and evaluate ICT solutions that integrate urban traffic management systems with the management of freight and logistics in urban areas. Finally, EURIDICE (European Inter-Disciplinary Research on Intelligent Cargo for Efficient, Safe and Environment-friendly Logistics) is a project sponsored by the European Commission under the 7th Framework Program seeking to develop an advanced European logistics system around the concept of 'intelligent cargo'. The goal is networking cargo objects like packages, vehicles and containers to provide information services whenever required along the transport chain. The project aims to build an information service platform centred on the individual cargo item and its interaction with the surrounding environment and the user.

3 URBAN TRAFFIC MANAGEMENT

Traffic congestion of urban roads undermines mobility in major cities. Traditionally, the congestion problem on surface streets was dealt by adding more lanes and new links to the existing Urban Traffic Networks (UTN). Since such a solution can no longer be considered for limited availability of space in urban centres, greater emphasis is nowadays placed on traffic management through the implementation and operation of ICT. In particular, traffic signal control on surface street networks plays a central role in traffic management. Despite the large research efforts on the topic, the problem of urban intersection congestion remains an open issue (Lo, 2001, Papageorgiou, 1999). Most of the currently implemented traffic control systems may be grouped into two principal classes (Papageorgiou et al., 2003, Patel and Ranganathan, 2001): i) fixed time strategies, that are derived off-line by use of optimization codes based on historical traffic data; ii) vehicle actuated strategies, that perform an on-line optimization and synchronization of the signal timing plans and make use of real time measurements. While the fixed time strategies do not use information on the actual traffic situation, the second actuated control class can be viewed as a traffic responsive network signal policy employing signal timing plans that respond automatically to traffic conditions. In a real time control strategy, detectors located on the intersection approaches monitor traffic conditions and feed information on the actual system state to the real time controller, which selects the duration of the phases in the signal timing plan in order to optimize an objective function. Although the corresponding optimal control problem may readily be formulated, its real time solution and realization in a control loop has to face several difficulties such as the size and the combinatorial nature of the optimization problem, the measurements of traffic conditions and the presence of unpredictable disturbances. The first and most notable of vehicle actuated techniques is the British SCOOT (Hunt et al., 1982), that decides an incremental change of splits, offsets and cycle times based on real time measurements. However, although SCOOT exhibits a centralized hardware architecture, the strategy is functionally decentralized with regard to splits setting. A formulation of the traffic signal network optimization strategy is presented in (Lo, 2001) and (Wey, 2000). However, the resulting procedures lead to complex mixed integer linear programming problems that are computationally intensive and the formulation for real networks requires heuristics for

solutions. Furthermore, Diakaki *et al.* (2002) propose a traffic responsive urban control strategy based on a feedback approach involving the application of a systematic and powerful control design method. Despite the simplicity and the efficiency of the proposed control strategy, such a modelling approach can not directly consider the effects of offset for consecutive junctions and the time-variance of the turning rates and the saturation flows.

An improvement on urban traffic actuated control strategies is provided in (Dotoli et al., 2006) where the green splits for a fixed cycle time are determined in real time, in order to minimize the number of vehicles in queue in the considered signalized area. The paper gives a contribution in facing the *apparently insurmountable difficulties* (Papageorgiou et al., 2003) in the real time solution and realization of the control loop governing an urban intersection by traffic lights. To this aim, the paper pursues simplicity in the modelling and in the optimization procedure by presenting a macroscopic model to describe the urban traffic network. Describing the system by a discrete time model with the sampling time equal to the cycle, the timing plan is obtained on the basis of the real traffic knowledge and the traffic measurements in a prefixed set of cycles. The traffic urban control strategy is performed by solving a mathematical programming problem that minimizes the number of vehicles in the considered urban area. The minimization of the objective function is subject to linear constraints derived from the intersection topology, the fixed cycle duration and the minimum and maximum duration of the phases commonly adopted in practice. The optimization problem is solved by a standard optimization software on a personal computer, so that practical applications are possible in a real time control framework.

4 INTERMODAL TRANSPORTATION NETWORKS

Intermodal Transportation Networks (ITN) are logistics systems integrating different transportation services, designed to move goods from origin to destination in a timely manner and using multiple modes of transportation (rail, ocean vessel, truck etc.). In the related literature several papers analyze ITN operations and planning issues as container fleet management, container terminal operations and scheduling. With the development of the new ICT tools, these operative and planning issues can be dealt with in a different way. In fact, these new technologies can effectively impact on the planning and operation of ITS. In particular, ICT solutions can increase the data flow and the information quality while allowing real-time data exchange in transportation systems (Crainic and Kim, 2007, Ramstedt and Woxenius, 2006). As mentioned in (Giannopoulos, 2004), numerous new applications of ICT to the transportation field are in various stages of development, but in the information transfer area the new systems seem to be too unimodal. In the application of ICT solutions to multimodal chains, an important and largely unexplored research field is the assessment of the impact of new technologies before their implementation, by a cost-benefit analysis (Zografos and Regan, 2004, Crainic and Kim, 2007). This research field offers numerous research opportunities: for instance, a not well explored case is that of coordinating independent stakeholders in the presence of uncertainties and lack of information on the stakeholders operations and their propagation within the intermodal chain.

An efficient ITN needs to synchronize the logistics operations. Therefore, information exchange among stakeholders is essential and ICT solutions are key tools to achieve efficiency. Nevertheless, the increasing complexity of these systems and the availability of the modern ICT for the interaction among the different decision makers and for the acquisition of information by the decision makers, require both the development of suitable models and the solution of new decision problems. Moreover, ITN and their decision making process are complex systems characterized by a high degree of interaction, concurrency and synchronization. Hence, ITS can be modeled as Discrete Event Systems (DES), whose dynamics depends on the interaction of discrete events, such as demands, departures and arrivals of means of transport at terminals and acquisitions and releases of resources by vehicles. DES models are widely used to describe decision making and operational processes. In the domain of ITN, the potentialities of these models are not fully explored and exploited. In particular at the operational level, we recall the models in the Petri net (Peterson, 1981) frameworks (Danielis et al., 2009, Di Febbraro et al., 2006, Fischer et al., 2000) and the simulation models (Boschian et al., 2009, Parola and Sciomachen, 2005).

In this presentation we mention two papers (Boschian et al., 2009) and (Danielis et al., 2009) that point out the role and the impact of the ICT applications in the field of the ITN management and control. In particular, paper (Danielis et al., 2009) focuses on the ICT solutions that allow sharing information among stakeholders on the basis of user friendly technologies. To this aim the authors single out some performance indices to evaluate activities, resources (utilization) and output (throughput, lead time) by integrating information flows allowed by the use of ICT tools. A case study is analyzed considering an ITN constituted by a port and a truck terminal of an Italian town including the road-ship transshipment process. The system is modeled and simulated in a timed Petri net framework considering different dynamic conditions characterized by a diverse level of information shared between terminals and operators. The simulation results show that ICT have a huge potential for efficient real time management and operation of ITN, as well as an effective impact on the infrastructures, reducing both the utilization of the system resources as well as the cost performance indices.

An application of the ICT tools to the real-time transport monitoring in order to trace and safely handle moving goods is presented in (Boschian et al., 2009). In particular, the authors analyze and simulate a real case study involving an ITN system and the transport and the customs clearance of goods that arrive to the port and the intermodal terminal. The case study is analyzed in the frame of the EURIDICE Integrated Project. The structure and the dynamics of the ITN model is described by the Unified Modeling Language formalism (Miles and Hamilton, 2006) and is implemented by a discrete-event simulation in Arena environment. The task is to provide services for the efficient utilization of infrastructures, both singularly and across territorial networks (e.g., port terminal synchronization with rail and road connections) and to contain the impact of logistic infrastructures on the local communities, reducing congestion and pollution caused by the associated freight movements. The discrete event simulation study shows that the application of the ICT tools allows us to locate goods and the related up-to-date information and to extend it with useful information-based services. Summing up, the simulation results show that integrating ICT into the system leads to a more efficient system management and drastically reduces the system lead times.

5 CONCLUSIONS

The paper presents the new attractive researches and projects in the field of ITS operational control and management. In particular, the key solutions of using effectively and efficiently the latest developments of ICT for ITS operational management are pointed out. The presentation focuses on the most important European Projects in ITS and on two crucial fields of the ITS management and control: the management of Urban Traffic networks and of Intermodal Transportation Networks. In the two cases are emphasized the new results and the challenges of future researches.

REFERENCES

Boschian, V., Fanti, M.P., Iacobellis, G., Ukovich, W., 2009. Using Information and Communication Technologies in Intermodal Freight Transportation: a Case Study. Submitted for publication.

Crainic, T.G., Kim, K.H., 2007 Intermodal transportation, In: C. Barnhart and G. Laporte, Editors, *Transportation, Handbooks in Operations Research and Management Science*, vol. 14, North-Holland, Amsterdam, pp. 467–537.

Danielis, R., Dotoli, M., Fanti, M.P., Mangini, A.M., Pesenti R., Stecco G., Ukovich W., 2009, "Integrating ICT into Logistics Intermodal Systems: A Petri Net Model of the Trieste Port", The European Control Conference 2009, ECC'09, August 23-26, Budapest, Hungary.

Diakaki, C., Papageorgiou, M., Aboudolas, K. 2002. A multivariable regulator approach to traffic-responsive network-wide signal control. *Control Engineering Practice*, *10*(2), 183-195.

Di Febbraro, A., Giglio, D., Sacco, N., 2002. On applying Petri nets to determine optimal offsets for coordinated traffic light timings. *Proc. 5th IEEE Int. Conf. on Intelligent Transportation Systems* (pp. 773-778), Singapore.

Di Febbraro, A., Giglio, D., Sacco, N., 2004. Urban traffic control structure based on hybrid Petri nets. *IEEE Trans. On Intelligent Transportation Systems 5,* (4), 224-237.

Di Febbraro, A., Porta, G., N. Sacco, N., 2006. A Petri net modelling approach of intermodal terminals based on Metrocargo system, *Proc. Intelligent Transportation Systems Conf.*, pp. 1442–1447.

Dotoli, M., Fanti, M.P., Meloni, C., 2006. A Signal Timing Plan Formulation for Urban Traffic Control. *Control Engineering Practice*, vol. 14, no.11, 2006, pp. 1297-1311.

Fischer, M., Kemper, P., 2000. Modeling and analysis of a freight terminal with stochastic Petri nets. In *Proc. 9th IFAC Symposium Control in Transportation Systems*.

Giannopoulos, G.A., 2004. The application of information and communication technologies in transport. In *European Journal Of Operational Research*, vol. 152, pp. 302-320.

Hunt, P.B., Robertson, D.L., Beterton, R.D., & Royle, M.C., 1982. The SCOOT on-line traffic signal optimization technique. *Traffic Engineering and Control, 23,* 190-199.

Lo, H. K., 2001. A cell-based traffic control formulation: strategies and benefits of dynamic timing plans. *Transportation Science*, *35*(2), 148-164.

Miles, R., Hamilton, K., 2006. *Learning UML 2.0.* O'Reilly Media, Sabastopol CA USA.

Papageorgiou, M., 1999. Automatic control methods in traffic and transportation. In *Operations research and decision aid methodologies in traffic and transportation management*, P. Toint, M. Labbe, K. Tanczos, & G. Laporte (Eds.), Springer-Verlag, 46-83.

Papageorgiou, M., Diakaki, C., Dinopoulou, V., Kotsialos, A., Wang, Y., 2003. Review of road traffic control strategies. *Proceedings of the IEEE*, *91*(12), 2043-2067.

Parola F., Sciomachen, A., 2005. Intermodal container flows in a port system metwor: Analysis of possible growths via simulation models. In *International Journal of Production Economics*, vol. 97, pp. 75-88.

Patel, M., Ranganathan, N., 2001. IDUTC: an intelligent decision-making system for urban traffic control applications. In *IEEE Trans. on Vehicular Technology*, *50*(3), 816-829.

Peterson, J.L., 1981. *Petri Net Theory and the Modeling of Systems*. Prentice Hall, Englewood Cliffs, NJ, USA.

Ramstedt, L., Woxenius, J., 2006. Modelling approaches to operational decision-making in freight transport chains, *Proc. 18th NOFOMA Conf.*.

Wey W.-M., 2000. Model formulation and solution algorithm of traffic signal control in an urban network. In *Computers, Environment and Urban Systems*, *24*(4), 355-377.

Zografos, K.G., Regan, A., 2004. Current Challenges for Intermodal Freight Transport and Logistics in Europe and the US. In *Journal of the Transportation Research Board*, vol. 1873, pp. 70-78.

BRIEF BIOGRAPHY

Maria Pia Fanti is associate professor in Systems and Control Engineering and is with the Department of Electrical and Electronic Engineering of the Polytechnic of Bari (Italy). Maria Pia Fanti received the Laurea degree in Electronic Engineering from the University of Pisa (Italy), in 1983 and obtained an IBM thesis award. She was a visiting researcher at the Rensselaer Polytechnic Institute of Troy, New York, in 1999. Her research interests include discrete event systems, Petri nets, modeling and control of automated manufacturing systems, modeling and management of logistics system, supply chains and health care systems.

Prof. Fanti is Associate Editor of the following journals: IEEE Trans. on Systems, Man, and Cybernetics. Part A, IEEE Trans. on Automation Science and Engineering, The Mediterranean J. of Measurement and Control, Int. J. of Automation and Control, and Enterprise Information Systems. She is Co-Chair of the Technical committee on Discrete Event Systems for the IEEE SMC Society, Chair of the Italy Section SMC Chapter, and member of the IFAC Technical Committee on Discrete Event and Hybrid Systems. She is authors of 120+ papers. She has served in 20+ conference international program committees, she is IPC chair of 2nd IFAC Workshop on Dependable Control of Discrete Systems, Bari, Italy, 2009 and of the IEEE Workshop on Health Care Management, Venice, Italy, 2010.

RECENT ADVANCES IN VERIFICATION AND ANALYSIS OF HYBRID SYSTEMS

Janan Zaytoon
University of Reims Champagne Ardennes
France

Abstract: Formal verification of properties is a very important area of analysis of hybrid systems. It is, indeed, essential to use methods and tools to guarantee that the global behaviour of a system is correct and consistent with the specifications. This is especially true for safety properties that insure that the system is not dangerous for itself or its environment.
Classically, verification of Safety properties may be performed with reachability computation in the hybrid state space. Basic ideas have not really evolved since the first works, however new techniques have been proposed and algorithms have been improved.
The aim of this talk is to present the problem of verification and reachability computation for hybrid systems and to propose a classification of recent improvements. To overcome the difficulties in verification and reachability analysis it is necessary to make choices regarding general principles, algorithms and mathematical representation of regions of the continuous state space. These choices depend on each other and must be consistent. However all approaches are based on common considerations that will be used to structure the talk.

BRIEF BIOGRAPHY

Born in 1962, Janan Zaytoon (BSc Eng./1983, MSc Eng./1986, DEA/1988, PhD/1993, Habilitation/1997) is Professor and Head of the CReSTIC Research Centre (involving 150 researchers) at Reims University. He was a member of the Administration Council of the same University (2003-2006). He is the Chair of the French national research network/group "GDR MACS of CNRS", which involves all the researchers in the field of Automatic Control Syetems in France (about 2000 researchers and PhD students).

His involvement in IFAC includes his service as member of the IFAC Council from 2008 to 2011, head of the French National Member Organizer since 1999, Chair of Technical Committee on Discrete Event and Hybrid Systems from 2005 to 2008, Vice-Chair of this Technical Committee from 2002 to 2005 and 2008 to 2011, member of the Publication Committee of IFAC from 2008 to 2011, Editor of the IFAC Journal "Control Engineering Practice" and the Affiliated IFAC Journal "Nonlinear Analysis: Hybrid Systems".

Professor Janan Zaytoon is the author/co-author of 70 journal papers, 3 books, 12 book chapters, 120 conference papers, and 8 patents. His main research interests are in the fields of Discrete Event Systems, Hybrid Dynamic Systems, Intelligent Control and Biomedical Engineering. He is an associate Editor of "IET Control Theory and Applications" and "Discrete Event Dynamic Systems", IPC and/or NOC Chair/Co-Chair of 15 Conferences, Editor/Co-Editor of 10 Conference Proceedings, Keynote speaker for 6 conferences, supervisor of 20 PhD students, Guest Editor/Co-editor for 18 special issues of 6 international and 2 national journals, leader of 8 industrial contracts, and was Chair of the WODES (International Workshop on Discrete Event Systems) steering Committee.

DISCRETE EVENT DIAGNOSIS USING PETRI NETS

Maria Paola Cabasino, Alessandro Giua and Carla Seatzu
Department of Electrical and Electronic Engineering, University of Cagliari, Piazza D'Armi, 09123 Cagliari, Italy
{*cabasino, giua, seatzu*}*@diee.unica.it*

Keywords: Petri nets, Diagnosis, Discrete event systems.

Abstract: This paper serves as a support for the plenary address given by the second author during the conference. In this paper we present an approach to on-line diagnosis of discrete event systems based on labeled Petri nets, that are a particular class of Petri nets where some events are undistinguishable, i.e., events that produce an output signal that is observable, but that is common to other events. Our approach is based on the notion of basis markings and justifications and it can be applied both to bounded and unbounded Petri nets whose unobservable subnet is acyclic. Moreover it is shown that, in the case of bounded Petri nets, the most burdensome part of the procedure may be moved off-line, computing a particular graph that we call *Basis Reachability Graph*.
Finally we present a diagnosis MATLAB toolbox with some examples of application.

1 INTRODUCTION

Failure detection and isolation in industrial systems is a subject that has received a lot of attention in the past few decades. A failure is defined to be any deviation of a system from its normal or intended behavior. Diagnosis is the process of detecting an abnormality in the system behavior and isolating the cause or the source of this abnormality.

Failures are inevitable in today's complex industrial environment and they could arise from several sources such as design errors, equipment malfunctions, operator mistakes, and so on. As technology advances, as we continue to build systems of increasing size and functionality, and as we continue to place increasing demands on the performance of these systems, then so do we increase the complexity of these systems. Consequently (and unfortunately), we enhance the potential for systems to fail, and no matter how safe our designs are, how improved our quality control techniques are, and how better trained the operators are, system failures become unavoidable.

Given the fact that failures are inevitable, the need for effective means of detecting them is quite apparent if we consider their consequences and impacts not just on the systems involved but on the society as a whole. Moreover we note that effective methods of failure diagnosis can not only help avoid the undesirable effects of failures, but can also enhance the operational goals of industries. Improved quality of performance, product integrity and reliability, and reduced cost of equipment maintenance and service are some major benefits that accurate diagnosis schemes can provide, especially for service and product oriented industries such as home and building environment control, office automation, automobile manufacturing, and semiconductor manufacturing. Thus, we see that accurate and timely methods of failure diagnosis can enhance the safety, reliability, availability, quality, and economy of industrial processes.

The need of automated mechanisms for the timely and accurate diagnosis of failures is well understood and appreciated both in industry and in academia. A great deal of research effort has been and is being spent in the design and development of automated diagnostic systems, and a variety of schemes, differing both in their theoretical framework and in their design and implementation philosophy, have been proposed.

In diagnosis approach two different problems can be solved: the problem of diagnosis and the problem of diagnosability.

Solving a problem of diagnosis means that we associate to each observed string of events a diagnosis state, such as "normal" or "faulty" or "uncertain". Solving a problem of diagnosability is equivalent to determine if the system is diagnosable, i.e., to determine if, once a fault has occurred, the system can detect its occurrence in a finite number of steps.

The diagnosis of discrete event systems (DES) is a research area that has received a lot of attention in the last years and has been motivated by the practical need of ensuring the correct and safe functioning of large complex systems. As discussed in the next session the first results have been presented within the framework of automata. More recently, the diagnosis problem has also been addressed using Petri nets (PNs). In fact, the use of Petri nets offers significant advantages because of their twofold representation: graphical and mathematical. Moreover, the intrinsically distributed nature of PNs where the notion of state (i.e., marking) and action (i.e., transition) is local reduces the computational complexity involved in solving a diagnosis problem.

In this paper we summarize our main contributions on diagnosis of DES using PNs (Giua and Seatzu, 2005; Cabasino et al., 2008; Lai et al., 2008; Cabasino et al., 2009). In particular, we focus on arbitrary labeled PNs where the observable events are the labels associated to transitions, while faults are modeled as silent transitions. We assume that there may also be transitions modeling a regular behavior, that are silent as well. Moreover, two or more transitions that may be simultaneously enabled may share the same label, thus they are undistinguishable. Our diagnosis approach is based on the definition of four diagnosis states modeling different degrees of alarm and it applies to all systems whose unobservable subnet is acyclic. Two are the main advantages of our procedure. First, we do not need an exhaustive enumeration of the states in which the system may be: this is due to the introduction of basis markings. Secondly, in the case of bounded net systems we can move off-line the most burdensome part of the procedure building a finite graph called basis reachability graph.

The paper is organized as follows. In Section 2 the state of art of diagnosis for discrete event systems is illustrated. In Section 3 we provide a background on PNs. In Sections 4 and 5 are introduced the definitions of minimal explanations, justifications and basis markings, that are the basic notions of our diagnosis approach. In Section 6 the diagnosis states are defined and a characterization of them in terms of basis markings and j-vectors is given. In Section 7 we show how the most burdensome part of the procedure can be moved offline in the case of bounded PNs. In Section 8 we present the MATLAB toolbox developed by our group for PNs diagnosis and in Section 9 we present some numerical results obtained applying our tool to a parametric model of manufacturing system. In Section 10 we draw the conclusions.

2 LITERATURE REVIEW

In this section we present the state of art of diagnosis of DES using automata and PNs.

2.1 Diagnosis of DES using Automata

In the contest of DES several original theoretical approaches have been proposed using *automata*.

In (Lin, 1994) and (Lin et al., 1993) a state-based DES approach to failure diagnosis is proposed. The problems of off-line and on-line diagnosis are addressed separately and notions of diagnosability in both of these cases are presented. The authors give an algorithm for computing a diagnostic control, i.e., a sequence of test commands for diagnosing system failures. This algorithm is guaranteed to converge if the system satisfies the conditions for on-line diagnosability.

In (Sampath et al., 1995) and (Sampath et al., 1996) the authors propose an approach to failure diagnosis where the system is modeled as a DES in which the failures are treated as unobservable events. The level of detail in a discrete event model appears to be quite adequate for a large class of systems and for a wide variety of failures to be diagnosed. The approach is applicable whenever failures cause a distinct change in the system status but do not necessarily bring the system to a halt. In (Sampath et al., 1995) a definition of diagnosability in the framework of formal languages is provided and necessary and sufficient conditions for diagnosability of systems are established. Also presented in (Sampath et al., 1995) is a systematic approach to solve the problem of diagnosis using diagnosers.

In (Sampath et al., 1998) the authors present an integrated approach to control and diagnosis. More specifically, authors present an approach for the design of diagnosable systems by appropriate design of the system controller and this approach is called active diagnosis. They formulate the active diagnosis problem as a supervisory control problem. The adopted procedure for solving the active diagnosis problem is the following: given the non-diagnosable language generated by the system of interest, they first select an “appropriate” sublanguage of this language as the legal language. Choice of the legal language is a design issue and typically depends on considerations such as acceptable system behavior (which ensures that the system behavior is not restricted more than necessary in order to eventually make it diagnosable) and detection delay for the failures. Once the appropriate legal language is chosen, they then design a controller (diagnostic controller), that achieves a

closed-loop language that is within the legal language and is diagnosable. This controller is designed based on the formal framework and the synthesis techniques that supervisory control theory provides, with the additional constraint of diagnosability.

In (Debouk et al., 2000) is addressed the problem of failure diagnosis in DES with decentralized information. Debouk *et al.* propose a coordinated decentralized architecture consisting of two local sites communicating with a coordinator that is responsible for diagnosing the failures occurring in the system. They extend the notion of diagnosability, originally introduced in (Sampath et al., 1995) for centralized systems, to the proposed coordinated decentralized architecture. In particular, they specify three protocols that realize the proposed architecture and analyze the diagnostic properties of these protocols.

In (Boel and van Schuppen, 2002) the authors address the problem of synthesizing communication protocols and failure diagnosis algorithms for decentralized failure diagnosis of DES with costly communication between diagnosers. The costs on the communication channels may be described in terms of bits and complexity. The costs of communication and computation force the trade-off between the control objective of failure diagnosis and that of minimization of the costs of communication and computation. The results of this paper is an algorithm for decentralized failure diagnosis of DES for the special case of only two diagnosers.

In (Zad et al., 2003) a state-based approach for on-line passive fault diagnosis is presented. In this framework, the system and the diagnoser (the fault detection system) do not have to be initialized at the same time. Furthermore, no information about the state or even the condition (failure status) of the system before the initiation of diagnosis is required. The design of the fault detection system, in the worst case, has exponential complexity. A model reduction scheme with polynomial time complexity is introduced to reduce the computational complexity of the design. Diagnosability of failures is studied, and necessary and sufficient conditions for failure diagnosability are derived.

2.2 Diagnosis of DES using Petri Nets

Among the first pioneer works dealing with PNs, we recall the approach of Prock. In (Prock, 1991) the author proposes an on-line technique for fault detection that is based on monitoring the number of tokens residing into P-invariants: when the number of tokens inside P-invariants changes, then the error is detected.

In (Sreenivas and Jafari, 1993) the authors employ time PNs to model the DES controller and backfiring transitions to determine whether a given state is invalid. Later on, time PNs have been employed in (Ghazel et al., 2005) to propose a monitoring approach for DES with unobservable events and to represent the "a priori" known behavior of the system, and track on-line its state to identify the events that occur.

In (Hadjicostis and Veghese, 1999) the authors use PN models to introduce redundancy into the system and additional P-invariants allow the detection and isolation of faulty markings.

Redundancy into a given PN is used in (Wu and Hadjicostis, 2005) to enable fault detection and identification using algebraic decoding techniques. In this paper Wu and Hadjicostis consider two types of faults: place faults that corrupt the net marking, and transition faults that cause a not correct update of the marking after event occurrence. Although this approach is general, the net marking has to be periodically observable even if unobservable events occur. Analogously, in (Lefebvre and Delherm, 2007) the authors investigate on the determination of the set of places that must be observed for the exact and immediate estimation of faults occurrence.

In (Ruiz-Beltràn et al., 2007) Interpreted PNs are employed to model the system behavior that includes both events and states partially observable. Based on the Interpreted PN model derived from an on-line methodology, a scheme utilizing a solution of a programming problem is proposed to solve the problem of diagnosis.

Note that, all papers in this topic assume that faults are modeled by unobservable transitions. However, while the above mentioned papers assume that the marking of certain places may be observed, a series of papers have been recently presented that are based on the assumption that no place is observable (Basile et al., 2008; Benveniste et al., 2003; Dotoli et al., 2008; Genc and Lafortune, 2007).

In particular, in (Genc and Lafortune, 2007) the authors propose a diagnoser on the basis of a modular approach that performs the diagnosis of faults in each module. Subsequently, the diagnosers recover the monolithic diagnosis information obtained when all the modules are combined into a single module that preserves the behavior of the underlying modular system. A communication system connects the different modules and updates the diagnosis information. Even if the approach does not avoid the state explosion problem, an improvement is obtained when the system can be modeled as a collection of PN modules coupled through common places.

The main advantage of the approaches in (Genc

and Lafortune, 2007) consists in the fact that, if the net is bounded, the diagnoser may be constructed off-line, thus moving off-line the most burdensome part of the procedure. Nevertheless, a characterization of the set of markings consistent with the actual observation is needed. Thus, large memory may be required.

An improvement in this respect has been given in (Benveniste et al., 2003; Basile et al., 2008; Dotoli et al., 2008).

In particular, in (Benveniste et al., 2003) a net unfolding approach for designing an on-line asynchronous diagnoser is used. The state explosion is avoided but the on-line computation can be high due to the on-line building of the PN structures by means of the unfolding.

In (Basile et al., 2008) the diagnoser is built on-line by defining and solving Integer Linear Programming (ILP) problems. Assuming that the fault transitions are not observable, the net marking is computed by the state equation and, if the marking has negative components, an unobservable sequence is occurred. The linear programming solution provides the sequence and detects the fault occurrences. Moreover, an off-line analysis of the PN structure reduces the computational complexity of the ILP problem.

In (Dotoli et al., 2008), in order to avoid the redesign and the redefinition of the diagnoser when the structure of the system changes, the authors propose a diagnoser that works on-line. In particular, it waits for an observable event and an algorithm decides whether the system behavior is normal or may exhibit some possible faults. To this aim, some ILP problems are defined and provide eventually the minimal sequences of unobservable transitions containing the faults that may have occurred. The proposed approach is a general technique since no assumption is imposed on the reachable state set that can be unlimited, and only few properties must be fulfilled by the structure of the PN modeling the system fault behavior.

We also proposed a series of contributions dealing with diagnosis of PNs (Giua and Seatzu, 2005; Cabasino et al., 2008; Lai et al., 2008; Cabasino et al., 2009). Our main results are summarized in the rest of the paper.

Note that none of the above mentioned papers regarding PNs deal with *diagnosability*, namely none of them provide a procedure to determine a priori if a system is *diagnosable*, i.e., if it is possible to reconstruct the occurrence of fault events observing words of finite length.

In fact, whereas this problem has been extensively studied within the framework of automata as discussed above, in the PN framework very few results have been presented.

The first contribution on diagnosability of PNs was given in (Ushio et al., 1998). They extend a necessary and sufficient condition for diagnosability in (Sampath et al., 1995; Sampath et al., 1996) to unbounded PN. They assume that the set of places is partitioned into observable and unobservable places, while all transitions are unobservable in the sense that their occurrences cannot be observed. Starting from the PN they build a diagnoser called *simple* ω *diagnoser* that gives them sufficient conditions for diagnosability of unbounded PNs.

In (Chung, 2005) the authors, in contrast with Ushio's paper, assumes that part of the transitions of the PN modelling is observable and shows as the additional information from observed transitions in general adds diagnosability to the analysed system. Moreover starting from the diagnoser he proposes an automaton called *verifier* that allows a polynomial check mechanism on diagnosability but for finite state automata models.

In (Wen and Jeng, 2005) the authors propose an approach to test diagnosability by checking the structure property of T-invariant of the nets. They use Ushio's diagnoser to prove that their method is correct, however they don't construct a diagnoser for the system to do diagnosis. In (Wen et al., 2005) they also present an algorithm, based on a linear programming problem, of polynomial complexity in the number of nodes for computing a sufficient condition of diagnosability of DES modeled by PN.

3 BACKGROUND

In this section we recall the formalism used in the paper. For more details on PNs we refer to (Murata, 1989).

A *Place/Transition net* (P/T net) is a structure $N = (P, T, Pre, Post)$, where P is a set of m places; T is a set of n transitions; $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ are the *pre–* and *post–* incidence functions that specify the arcs; $C = Post - Pre$ is the incidence matrix.

A *marking* is a vector $M : P \to \mathbb{N}$ that assigns to each place of a P/T net a non–negative integer number of tokens, represented by black dots. We denote $M(p)$ the marking of place p. A P/T *system* or *net system* $\langle N, M_0 \rangle$ is a net N with an initial marking M_0. A transition t is enabled at M iff $M \geq Pre(\cdot, t)$ and may fire yielding the marking $M' = M + C(\cdot, t)$. We write $M[\sigma\rangle$ to denote that the sequence of transitions $\sigma = t_{j_1} \cdots t_{j_k}$ is enabled at M, and we write $M[\sigma\rangle M'$ to denote that the firing of σ yields M'. We also write $t \in \sigma$ to denote that a transition t is contained in σ.

The set of all sequences that are enabled at the initial marking M_0 is denoted $L(N,M_0)$, i.e., $L(N,M_0) = \{\sigma \in T^* \mid M_0[\sigma\rangle\}$.

Given a sequence $\sigma \in T^*$, we call $\pi : T^* \to \mathbb{N}^n$ the function that associates to σ a vector $y \in \mathbb{N}^n$, named the *firing vector* of σ. In particular, $y = \pi(\sigma)$ is such that $y(t) = k$ if the transition t is contained k times in σ.

A marking M is *reachable* in $\langle N,M_0\rangle$ iff there exists a firing sequence σ such that $M_0\ [\sigma\rangle\ M$. The set of all markings reachable from M_0 defines the *reachability set* of $\langle N,M_0\rangle$ and is denoted $R(N,M_0)$.

A PN having no directed circuits is called *acyclic*. A net system $\langle N,M_0\rangle$ is *bounded* if there exists a positive constant k such that, for $M \in R(N,M_0)$, $M(p) \leq k$.

A *labeling function* $\mathcal{L} : T \to L \cup \{\varepsilon\}$ assigns to each transition $t \in T$ either a symbol from a given alphabet L or the empty string ε.

We denote as T_u the set of transitions whose label is ε, i.e., $T_u = \{t \in T \mid \mathcal{L}(t) = \varepsilon\}$. Transitions in T_u are called *unobservable* or *silent*. We denote as T_o the set of transitions labeled with a symbol in L. Transitions in T_o are called *observable* because when they fire their label can be observed. Note that in this paper we assume that the same label $l \in L$ can be associated to more than one transition. In particular, two transitions $t_1, t_2 \in T_o$ are called *undistinguishable* if they share the same label, i.e., $\mathcal{L}(t_1) = \mathcal{L}(t_2)$. The set of transitions sharing the same label l are denoted as T_l.

In the following we denote as C_u (C_o) the restriction of the incidence matrix to T_u (T_o) and denote as n_u and n_o, respectively, the cardinality of the above sets. Moreover, given a sequence $\sigma \in T^*$, $P_u(\sigma)$, resp., $P_o(\sigma)$, denotes the projection of σ over T_u, resp., T_o.

We denote as w the word of events associated to the sequence σ, i.e., $w = P_o(\sigma)$. Note that the length of a sequence σ (denoted $|\sigma|$) is always greater than or equal to the length of the corresponding word w (denoted $|w|$). In fact, if σ contains k' transitions in T_u then $|\sigma| = k' + |w|$.

Definition 3.1 (Cabasino et al., 2009). Let $\langle N,M_0\rangle$ be a labeled net system with labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$, where $N = (P,T,Pre,Post)$ and $T = T_o \cup T_u$. Let $w \in L^*$ be an observed word. We define

$$\mathcal{S}(w) = \{\sigma \in L(N,M_0) \mid P_o(\sigma) = w\}$$

the set of firing sequences *consistent* with $w \in L^*$, and

$$\mathcal{C}(w) = \{M \in \mathbb{N}^m \mid \exists \sigma \in T^* : P_o(\sigma) = w \wedge M_0[\sigma\rangle M\}$$

the set of markings *consistent* with $w \in L^*$. ■

In plain words, given an observation w, $\mathcal{S}(w)$ is the set of sequences that may have fired, while $\mathcal{C}(w)$ is the set of markings in which the system may actually be.

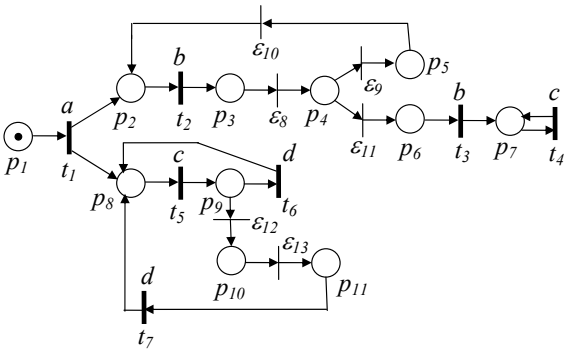


Figure 1: A PN system modeling.

Example 3.2. Let us consider the PN in Figure 1. Let us assume $T_o = \{t_1,t_2,t_3,t_4,t_5,t_6,t_7\}$ and $T_u = \{\varepsilon_8,\varepsilon_9,\varepsilon_{10},\ \varepsilon_{11},\varepsilon_{12},\varepsilon_{13}\}$, where for a better understanding unobservable transitions have been denoted ε_i rather than t_i. The labeling function is defined as follows: $\mathcal{L}(t_1) = a$, $\mathcal{L}(t_2) = \mathcal{L}(t_3) = b$, $\mathcal{L}(t_4) = \mathcal{L}(t_5) = c$, $\mathcal{L}(t_6) = \mathcal{L}(t_7) = d$.

First let us consider $w = ab$. The set of firing sequences that is consistent with w is $\mathcal{S}(w) = \{t_1t_2,\ t_1t_2\varepsilon_8,t_1t_2\varepsilon_8\varepsilon_9,t_1t_2\varepsilon_8\varepsilon_9\varepsilon_{10},t_1t_2\varepsilon_8\varepsilon_{11}\}$, and the set of markings consistent with w is $\mathcal{C}(w) = \{[0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T,[0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T,$ $[0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0]^T,[0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T,$ $[0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0]^T\}$.

If we consider $w = acd$ the set of firing sequences that are consistent with w is $\mathcal{S}(w) = \{t_1t_5t_6,t_1t_5\varepsilon_{12}\varepsilon_{13}t_7\}$, and the set of markings consistent with w is $\mathcal{C}(w) = \{[0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T\}$. Thus two different firing sequences may have fired (the second one also involving silent transitions), but they both lead to the same marking. ■

4 MINIMAL EXPLANATIONS AND MINIMAL E-VECTORS

In this section we present the notions of minimal explanations and minimal e-vectors for labeled PNs. First we introduce notions of explanations for unlabeled PNs, secondly we define when an explanation is minimal and finally we extend these concepts to labeled PN.

Definition 4.1 (Cabasino et al., 2008). Given a marking M and an observable transition $t \in T_o$, we define

$$\Sigma(M,t) = \{\sigma \in T_u^* \mid M[\sigma\rangle M',\ M' \geq Pre(\cdot,t)\}$$

the set of *explanations* of t at M, and

$$Y(M,t) = \pi(\Sigma(M,t))$$

the *e-vectors* (or *explanation vectors*), i.e., firing vectors associated to the explanations. ■

Thus $\Sigma(M,t)$ is the set of unobservable sequences whose firing at M enables t. Among the above sequences we want to select those whose firing vector is minimal. The firing vector of these sequences are called *minimal e-vectors*.

Definition 4.2 (Cabasino et al., 2008). Given a marking M and a transition $t \in T_o$, we define

$$\Sigma_{\min}(M,t) = \{\sigma \in \Sigma(M,t) \mid \nexists\, \sigma' \in \Sigma(M,t) : \pi(\sigma') \lneqq \pi(\sigma)\}$$

the set of *minimal explanations* of t at M, and we define

$$Y_{\min}(M,t) = \pi(\Sigma_{\min}(M,t))$$

the corresponding set of *minimal e-vectors*. ■

In (Corona et al., 2004) we proved that, if the unobservable subnet is acyclic and backward conflict-free, then $|Y_{\min}(M,t)| = 1$.

Different approaches can be used to compute $Y_{\min}(M,t)$, e.g., (Boel and Jiroveanu, 2004; Jiroveanu and Boel, 2004). In (Cabasino et al., 2008) we suggested an approach that terminates finding all vectors in $Y_{\min}(M,t)$ if applied to nets whose unobservable subnet is acyclic. It simply requires algebraic manipulations, and is inspired by the procedure proposed in (Martinez and Silva, 1982) for the computation of minimal P-invariants. For the sake of brevity, this algorithm is not reported here.

In the case of labeled PNs what we observe are symbols in L. Thus, it is useful to compute the following sets.

Definition 4.3 (Cabasino et al., 2009). Given a marking M and an observation $l \in L$, we define the set of *minimal explanations of l at M* as

$$\hat{\Sigma}_{\min}(M,l) = \cup_{t \in T_l} \cup_{\sigma \in \Sigma_{\min}(M,t)} (t,\sigma),$$

i.e., the set of pairs (transition labeled l; corresponding minimal explanation), and we define the set of *minimal e-vectors of l at M* as

$$\hat{Y}_{\min}(M,l) = \cup_{t \in T_l} \cup_{e \in Y_{\min}(M,t)} (t,e),$$

i.e., the set of pairs (transition labeled l; corresponding minimal e-vector). ■

Thus, $\hat{\Sigma}_{\min}(M,l)$ is the set of pairs whose first element is the transition labeled l and whose second element is the corresponding minimal explanation $\sigma \in \Sigma_{\min}(M,t)$, namely the corresponding sequence of unobservable transitions whose firing at M enables l and whose firing vector is minimal. Moreover, $\hat{Y}_{\min}(M,l)$ is the set of pairs whose first element is the transition labeled l and whose second element is the firing vector $e \in Y_{\min}(M,t)$ corresponding to the second element in $\hat{\Sigma}_{\min}(M,l)$.

Obviously, $\hat{\Sigma}_{\min}(M,l)$ and $\hat{Y}_{\min}(M,l)$ are a generalization of the sets of minimal explanations and minimal e-vectors introduced for unlabeled PNs with unobservable transitions. Moreover, in the above sets $\hat{\Sigma}_{\min}(M,l)$ and $\hat{Y}_{\min}(M,l)$ different sequences σ and different e-vectors e, respectively, are associated in general to the same $t \in T_l$.

5 BASIS MARKINGS AND J-VECTORS

In this section we introduce the definitions of basis markings and justifications that are the crucial notions of our diagnosis approach.

In particular, given a sequence of observed events $w \in L^*$, a basis marking M_b is a marking reached from M_0 with the firing of the observed word w and of all unobservable transitions whose firing is necessary to enable w. Note that, in general several sequences $\sigma_o \in T_o^*$ may correspond to the same w, i.e., there are several sequences of observable transitions such that $\mathcal{L}(\sigma_o) = w$ that may have actually fired. Moreover, in general, to any of such sequences σ_o a different sequence of unobservable transitions interleaved with it is necessary to make it firable at the initial marking. Thus we need to introduce the following definition of pairs (sequence of transitions in T_o labeled w; corresponding *justification*).

Definition 5.1 (Cabasino et al., 2009). Let $\langle N, M_0 \rangle$ be a net system with labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$, where $N = (P,T,Pre,Post)$ and $T = T_o \cup T_u$. Let $w \in L^*$ be a given observation. We define

$$\hat{\mathcal{J}}(w) = \{ (\sigma_o,\sigma_u),\ \sigma_o \in T_o^*,\ \mathcal{L}(\sigma_o) = w,\ \sigma_u \in T_u^* \mid [\exists \sigma \in \mathcal{S}(w) : \sigma_o = P_o(\sigma),\ \sigma_u = P_u(\sigma)] \wedge [\nexists \sigma' \in \mathcal{S}(w) : \sigma_o = P_o(\sigma'),\ \sigma'_u = P_u(\sigma') \wedge \pi(\sigma'_u) \lneqq \pi(\sigma_u)]\}$$

the set of pairs (sequence $\sigma_o \in T_o^*$ with $\mathcal{L}(\sigma_o) = w$; corresponding *justification* of w). Moreover, we define

$$\hat{Y}_{\min}(M_0,w) = \{(\sigma_o,y),\ \sigma_o \in T_o^*, \mathcal{L}(\sigma_o) = w, y \in \mathbb{N}^{n_u} \mid \exists (\sigma_o,\sigma_u) \in \hat{\mathcal{J}}(w) : \pi(\sigma_u) = y\}$$

the set of pairs (sequence $\sigma_o \in T_o^*$ with $\mathcal{L}(\sigma_o) = w$; corresponding *j-vector*). ■

In simple words, $\hat{\mathcal{J}}(w)$ is the set of pairs whose first element is the sequence $\sigma_o \in T_o^*$ labeled w and whose second element is the corresponding sequence of unobservable transitions interleaved with σ_o whose firing enables σ_o and whose firing vector is minimal.

The firing vectors of these sequences are called *j-vectors*.

Definition 5.2 (Cabasino et al., 2009). Let $\langle N, M_0 \rangle$ be a net system with labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$, where $N = (P, T, Pre, Post)$ and $T = T_o \cup T_u$. Let w be a given observation and $(\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(w)$ be a generic pair (sequence of observable transitions labeled w; corresponding minimal justification). The marking

$$M_b = M_0 + C_u \cdot y + C_o \cdot y', \qquad y = \pi(\sigma_u),\ y' = \pi(\sigma_o),$$

i.e., the marking reached firing σ_o interleaved with the minimal justification σ_u, is called *basis marking* and y is called its *j-vector* (or *justification-vector*). ■

Obviously, because in general more than one justification exists for a word w (the set $\hat{\mathcal{J}}(w)$ is generally not a singleton), the basis marking may be not unique as well.

Definition 5.3 (Cabasino et al., 2009). Let $\langle N, M_0 \rangle$ be a net system with labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$, where $N = (P, T, Pre, Post)$ and $T = T_o \cup T_u$. Let $w \in L^*$ be an observed word. We define

$$\begin{aligned}\mathcal{M}(w) = \{(M, y) \mid\ & (\exists \sigma \in \mathcal{S}(w) \ :\ M_0[\sigma\rangle M) \wedge \\ & (\exists (\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(w) \ :\ \sigma_o = P_o(\sigma), \\ & \sigma_u = P_u(\sigma),\ y = \pi(\sigma_u))\}\end{aligned}$$

the set of pairs (basis marking; relative j-vector) that are *consistent* with $w \in L^*$. ■

Note that the set $\mathcal{M}(w)$ does not keep into account the sequences of observable transitions that may have actually fired. It only keeps track of the basis markings that can be reached and of the firing vectors relative to sequences of unobservable transitions that have fired to reach them. Indeed, this is the information really significant when performing diagnosis. The notion of $\mathcal{M}(w)$ is fundamental to provide a recursive way to compute the set of minimal explanations.

Proposition 5.4 (Cabasino et al., 2009). Given a net system $\langle N, M_0 \rangle$ with labeling function $\mathcal{L} : T \to L \cup \{\varepsilon\}$, where $N = (P, T, Pre, Post)$ and $T = T_o \cup T_u$. Assume that the unobservable subnet is acyclic. Let $w = w'l$ be a given observation.

The set $\hat{Y}_{\min}(M_0, wl)$ is defined as:

$$\begin{aligned}\hat{Y}_{\min}(M_0, wl) = \quad & \{(\sigma_o, y) \mid \sigma_o = \sigma_0' t \wedge y = y' + e \ : \\ & (\sigma_o', y') \in \hat{Y}_{\min}(M_0, w), \\ & (t, e) \in \hat{Y}_{\min}(M_b', l) \text{ and } \mathcal{L}(t) = l\},\end{aligned}$$

where $M_b' = M_0 + C_u \cdot y' + C_o \cdot \sigma_o'$.

Example 5.5. Let us consider the PN in Figure 1 previously introduced in Example 3.2.

Let us assume $w = acd$. The set of justifications is $\hat{\mathcal{J}}(w) = \{(t_1 t_5 t_6, \varepsilon), (t_1 t_5 t_7, \varepsilon_{12}\varepsilon_{13})\}$ and the set of j-vectors is $\hat{Y}_{min}(M_0, w) = \{(t_1 t_5 t_6, \vec{0}), (t_1 t_5 t_7, [0\ 0\ 0\ 0\ 1\ 1]^T)\}$. The above j-vectors lead to the same basis marking $M_b = [0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T$ thus $\mathcal{M}(w) = \{(M_b, \vec{0}), (M_b, [0\ 0\ 0\ 0\ 1\ 1]^T)\}$.

Now, let us consider $w = ab$. In this case $\hat{\mathcal{J}}(w) = \{(t_1 t_2, \varepsilon)\}$, $\hat{Y}_{min}(M_0, w) = \{(t_1 t_2, \vec{0})\}$ and the basis marking is the same as in the previous case, namely $M_b = [0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T$, thus $\mathcal{M}(w) = \{(M_b, \vec{0})\}$. ■

Under the assumption of acyclicity of the unobservable subnet, the set $\mathcal{M}(w)$ can be easily constructed as follows.

Algorithm 5.6 (Computation of the basis markings and j-vectors).

1. Let $w = \varepsilon$.
2. Let $\mathcal{M}(w) = \{(M_0, \vec{0})\}$.
3. Wait until a new label l is observed.
4. Let $w' = w$ and $w = w'l$.
5. Let $\mathcal{M}(w) = \emptyset$.
6. For all M' such that $(M', y') \in \mathcal{M}(w')$, do
  6.1. for all $t \in T_l$, do
    6.1.1. for all $e \in Y_{\min}(M', t)$, do
    6.1.1.1. let $M = M' + C_u \cdot e + C(\cdot, t)$,
    6.1.1.2. for all y' such that $(M', y') \in \mathcal{M}(w')$, do
      6.1.2.1. let $y = y' + e$,
      6.1.2.2. let $\mathcal{M}(w) = \mathcal{M}(w) \cup \{(M, y)\}$.
7. Goto step 3.

■

In simple words, the above algorithm can be explained as follows. We assume that a certain word w (that is equal to the empty string at the initial step) has been observed. Then, a new observable t fires and we observe its label $\mathcal{L}(t)$ (e.g., l). We consider all basis markings at the observation w' before the firing of t, and we select among them those that may have allowed the firing of at least one transition $t \in T_l$, also taking into account that this may have required the firing of appropriate sequences of unobservable transitions. In particular, we focus on the minimal explanations, and thus on the corresponding minimal e-vectors (step 6.1.1). Finally, we update the set $\mathcal{M}(w)$ including all pairs of new basis markings and j-vectors, taking into account that for each basis marking at w' it may correspond more than one j-vector.

Let us now recall the following result.

Definition 5.7 (Cabasino et al., 2008). Let $\langle N, M_0 \rangle$ be a net system where $N = (P, T, Pre, Post)$ and $T = T_o \cup T_u$. Assume that the unobservable subnet is acyclic. Let $w \in T_o^*$ be an observed word. We denote

$$\mathcal{M}_{basis}(w) = \{M \in \mathbb{N}^m \mid \exists y \in \mathbb{N}^{n_u} \text{ and } (M, y) \in \mathcal{M}(w)\}$$

the set of basis markings at w. Moreover, we denote as

$$\mathcal{M}_{basis} = \bigcup_{w \in T_o^*} \mathcal{M}_{basis}(w)$$

the set of all basis markings for any observation w. ■

Note that if the net system is bounded then the set $\mathcal{M}_{basis}$ is *finite* being the set of basis markings a subset of the reachability set.

Theorem 5.8 (Cabasino et al., 2008). Let us consider a net system $\langle N, M_0 \rangle$ whose unobservable subnet is acyclic. For any $w \in L^*$ it holds that

$$\mathcal{C}(w) = \{M \in \mathbb{N}^m \mid M = M_b + C_u \cdot y : y \geq \vec{0} \text{ and } M_b \in \mathcal{M}_{basis}(w)\}.$$

6 DIAGNOSIS USING PETRI NETS

Assume that the set of unobservable transitions is partitioned into two subsets, namely $T_u = T_f \cup T_{reg}$ where T_f includes all fault transitions (modeling anomalous or fault behavior), while T_{reg} includes all transitions relative to unobservable but regular events. The set T_f is further partitioned into r different subsets T_f^i, where $i = 1, \ldots, r$, that model the different fault classes.

The following definition introduces the notion of *diagnoser*.

Definition 6.1 (Cabasino et al., 2009). A *diagnoser* is a function $\Delta : L^* \times \{T_f^1, T_f^2, \ldots, T_f^r\} \to \{0,1,2,3\}$ that associates to each observation $w \in L^*$ and to each fault class T_f^i, $i = 1, \ldots, r$, a *diagnosis state*.

- $\Delta(w, T_f^i) = 0$ if for all $\sigma \in \mathcal{S}(w)$ and for all $t_f \in T_f^i$ it holds $t_f \notin \sigma$.
  In such a case the ith fault cannot have occurred, because none of the firing sequences consistent with the observation contains fault transitions of class i.
- $\Delta(w, T_f^i) = 1$ if:
  (i) there exist $\sigma \in \mathcal{S}(w)$ and $t_f \in T_f^i$ such that $t_f \in \sigma$ but
  (ii) for all $(\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(w)$ and for all $t_f \in T_f^i$ it holds that $t_f \notin \sigma_u$.
  In such a case a fault transition of class i may have occurred but is not contained in any justification of w.
- $\Delta(w, T_f^i) = 2$ if there exist $(\sigma_o, \sigma_u), (\sigma_o', \sigma_u') \in \hat{\mathcal{J}}(w)$ such that
  (i) there exists $t_f \in T_f^i$ such that $t_f \in \sigma_u$;
  (ii) for all $t_f \in T_f^i$, $t_f \notin \sigma_u'$.
  In such a case a fault transition of class i is contained in one (but not in all) justification of w.
- $\Delta(w, T_f^i) = 3$ if for all $\sigma \in \mathcal{S}(w)$ there exists $t_f \in T_f^i$ such that $t_f \in \sigma$.
  In such a case the ith fault must have occurred, because all firable sequences consistent with the observation contain at least one fault in T_f^i. ■

Example 6.2. Let us consider the PN in Figure 1 previously introduced in Example 3.2. Let $T_f = \{\varepsilon_{11}, \varepsilon_{12}\}$. Assume that the two fault transitions belong to different fault classes, i.e., $T_f^1 = \{\varepsilon_{11}\}$ and $T_f^2 = \{\varepsilon_{12}\}$.

Let us observe $w = a$. Then $\Delta(w, T_f^1) = \Delta(w, T_f^2) = 0$, being $\hat{\mathcal{J}}(w) = \{(t_1, \varepsilon)\}$ and $\mathcal{S}(w) = \{t_1\}$. In words no fault of both fault classes can have occurred.

Let us observe $w = ab$. Then $\Delta(w, T_f^1) = 1$ and $\Delta(w, T_f^2) = 0$, being $\hat{\mathcal{J}}(w) = \{(t_1 t_2, \varepsilon)\}$ and $\mathcal{S}(w) = \{t_1 t_2, t_1 t_2 \varepsilon_8, t_1 t_2 \varepsilon_8 \varepsilon_9, t_1 t_2 \varepsilon_8 \varepsilon_9 \varepsilon_{10}, t_1 t_2 \varepsilon_8 \varepsilon_{11}\}$. This means that a fault of the second fault class may have occurred (e.g. $t_1 t_2 \varepsilon_8 \varepsilon_{11}$) but it is not contained in any justification of ab, while no fault of the first fault class can have occurred.

Now, let us consider $w = abb$. In this case $\Delta(w, T_f^1) = 2$ and $\Delta(w, T_f^2) = 0$, being $\hat{\mathcal{J}}(w) = \{(t_1 t_2 t_2, \varepsilon_8 \varepsilon_9 \varepsilon_{10}), (t_1 t_2 t_3, \varepsilon_8 \varepsilon_{11})\}$ and $\mathcal{S}(w) = \{t_1 t_2 \varepsilon_8 \varepsilon_9 \varepsilon_{10} t_2, t_1 t_2 \varepsilon_8 \varepsilon_9 \varepsilon_{10} t_2 \varepsilon_8, t_1 t_2 \varepsilon_8 \varepsilon_9 \varepsilon_{10} t_2 \varepsilon_8 \varepsilon_9, t_1 t_2 \varepsilon_8 \varepsilon_9 \varepsilon_{10} t_2 \varepsilon_8 \varepsilon_9 \varepsilon_{10}, t_1 t_2 \varepsilon_8 \varepsilon_9 \varepsilon_{10} t_2 \varepsilon_8 \varepsilon_{11}\}$. This means that no fault of the first fault class can have occurred, while a fault of the second fault class may have occurred since one justification does not contain ε_{11} and one justification contains it.

Finally, let us consider $w = abbccc$. In this case $\Delta(w, T_f^1) = 1$ and $\Delta(w, T_f^2) = 3$. In fact since $\hat{\mathcal{J}}(w) = \{(t_1 t_2 t_3 t_5 t_4 t_4, \varepsilon_8 \varepsilon_{11}), (t_1 t_2 t_3 t_4 t_5 t_4, \varepsilon_8 \varepsilon_{11}), (t_1 t_2 t_3 t_4 t_4 t_5, \varepsilon_8 \varepsilon_{11}), (t_1 t_2 t_3 t_4 t_4 t_4, \varepsilon_8 \varepsilon_{11})\}$ a fault of the first fault class must have occurred, while a fault of the second fault class may have occurred (e.g. $t_1 t_2 \varepsilon_8 \varepsilon_{11} t_3 t_4 t_4 t_5 \varepsilon_{12}$) but it is not contained in any justification of w. ■

The following proposition presents how the diagnosis states can be characterized analyzing basis markings and justifications.

Proposition 6.3 (Cabasino et al., 2009). Consider an observed word $w \in L^*$.

- $\Delta(w, T_f^i) \in \{0,1\}$ iff for all $(M, y) \in \mathcal{M}(w)$ and for all $t_f \in T_f^i$ it holds $y(t_f) = 0$.
- $\Delta(w, T_f^i) = 2$ iff there exist $(M, y) \in \mathcal{M}(w)$ and $(M', y') \in \mathcal{M}(w)$ such that:
  (i) there exists $t_f \in T_f^i$ such that $y(t_f) > 0$,
  (ii) for all $t_f \in T_f^i$, $y'(t_f) = 0$.

- $\Delta(w,T_f^i)=3$ iff for all $(M,y)\in\mathcal{M}(w)$ there exists $t_f\in T_f^i$ such that $y(t_f)>0$.

The following proposition shows how to distinguish between diagnosis states 0 and 1.

Proposition 6.4 (Cabasino et al., 2009). For a PN whose unobservable subnet is acyclic, let $w\in L^*$ be an observed word such that for all $(M,y)\in\mathcal{M}(w)$ it holds $y(t_f)=0\ \forall\ t_f\in T_f^i$. Let us consider the constraint set

$$\mathcal{T}(M) \quad = \quad \begin{cases} M+C_u\cdot z\geq\vec{0}, \\ \sum\limits_{t_f\in T_f^i} z(t_f)>0, \\ z\in\mathbb{N}^{n_u}. \end{cases} \tag{1}$$

- $\Delta(w,T_f^i)=0$ if $\forall\ (M,y)\in\mathcal{M}(w)$ the constraint set (1) is not feasible.
- $\Delta(w,T_f^i)=1$ if $\exists\ (M,y)\in\mathcal{M}(w)$ such that the constraint set (1) is feasible.

On the basis of the above two results, if the unobservable subnet is acyclic, diagnosis may be carried out by simply looking at the set $\mathcal{M}(w)$ for any observed word w and, should the diagnosis state be either 0 or 1, by additionally evaluating whether the corresponding integer constraint set (1) admits a solution.

Example 6.5. Let us consider the PN in Figure 1 where $T_f^1=\{\varepsilon_{11}\}$ and $T_f^2=\{\varepsilon_{12}\}$.

Let $w=ab$. In this case $\mathcal{M}(w)=\{(M_b^1,\vec{0})\}$, where $M_b^1=[0\,1\,0\,0\,0\,0\,0\,1\,0\,0\,0]^T$. Being $\mathcal{T}(M_b^1)$ feasible only for the fault class T_f^1 it holds $\Delta(w,T_f^1)=1$ and $\Delta(w,T_f^2)=0$.

Let $w=abb$. It is $\mathcal{M}(w)=\{(M_b^1,[1\ 1\ 1\ 0\ 0\ 0]^T),(M_b^2,[1\ 0\ 0\ 1\ 0\ 0]^T)\}$, where $M_b^2=[0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0]^T$. It is $\Delta(w,T_f^1)=2$ and $\Delta(w,T_f^2)=0$ being both $\mathcal{T}(M_b^1)$ and $\mathcal{T}(M_b^2)$ not feasible.

Let $w=abbccc$. In this case $\mathcal{M}(w)=\{(M_b^3,[1\ 1\ 1\ 0\ 0\ 0]^T),(M_b^4,[1\ 1\ 1\ 0\ 0\ 0]^T)\}$, where $M_b^3=[0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0]^T$ and $M_b^4=[0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0]^T$. It is $\Delta(w,T_f^1)=3$ and being $\mathcal{T}(M_b^4)$ feasible for the second fault class T_f^2 it holds $\Delta(w,T_f^2)=1$. ■

7 BASIS REACHABILITY GRAPH

Diagnosis approach described in the previous section can be applied both to bounded and unbounded PNs. The proposed approach is an on-line approach that for each new observed event updates the diagnosis state for each fault class computing the set of basis markings and j-vectors. Moreover if for a given fault class is necessary to distinguish between diagnosis states 0 and 1, it is also necessary to solve for each basis marking M_b the constraint set $\mathcal{T}(M_b)$.

In this section we show that if the considered net system is bounded, the most burdensome part of the procedure can be moved off-line defining a graph called *Basis Reachability Graph* (BRG).

Definition 7.1. The BRG is a deterministic graph that has as many nodes as the number of possible basis markings.

To each node is associated a different basis marking M and a row vector with as many entries as the number of fault classes. The entries of this vector may only take binary values: 1 if $\mathcal{T}(M)$ is feasible, 0 otherwise.

Arcs are labeled with observable events in L and e-vectors. More precisely, an arc exists from a node containing the basis marking M to a node containing the basis marking M' if and only if there exists a transition t for which an explanation exists at M and the firing of t and one of its minimal explanations leads to M'. The arc going from M to M' is labeled $(\mathcal{L}(t),e)$, where $e\in Y_{\min}(M,t)$ and $M'=M+C_u\cdot e+C(\cdot,t)$. ■

Note that the number of nodes of the BRG is always finite being the set of basis markings a subset of the set of reachable markings, that is finite being the net bounded. Moreover, the row vector of binary values associated to the nodes of the BRG allows us to distinguish between the diagnosis state 1 or 0.

The main steps for the computation of the BRG in the case of labeled PNs are summarized in the following algorithm.

Algorithm 7.2 (Computation of the BRG).

1. Label the initial node (M_0,x_0) where $\forall i=1,\ldots,r$,
$x_0(T_f^i)=\begin{cases}1 & \text{if } \mathcal{T}(M_0) \text{ is feasible,}\\ 0 & \text{otherwise.}\end{cases}$
Assign no tag to it.
2. While nodes with no tag exist
select a node with no tag and do
2.1. let M be the marking in the node (M,x),
2.2. for all $l\in L$
2.2.1. for all $t: L(t)=l\wedge Y_{\min}(M,t)\neq\emptyset$, do
- for all $e\in Y_{\min}(M,t)$, do
  - let $M'=M+C_u\cdot e+C(\cdot,t)$,
  - if $\nexists$ a node (M,x) with $M=M'$, do
    - add a new node to the graph containing (M',x') where $\forall i=1,\ldots,r$,
      $x'(T_f^i)=\begin{cases}1 & \text{if } \mathcal{T}(M') \text{ is feasible,}\\ 0 & \text{otherwise.}\end{cases}$
      and arc (l,e) from (M,x) to (M',x')
  - else
    - add arc (l,e) from (M,x) to (M',x')

if it does not exist yet
2.3. tag the node "old".
3. Remove all tags.

■

The algorithm constructs the BRG starting from the initial node to which it corresponds the initial marking and a binary vector defining which classes of faults may occur at M_0. Now, we consider all the labels $l \in L$ such that there exists a transition t with $L(t) = l$ for which a minimal explanation at M_0 exists. For any of these transitions we compute the marking resulting from firing t at $M_0 + C_u \cdot e$, for any $e \in Y_{\min}(M_0, t)$. If a pair (marking, binary vector) not contained in the previous nodes is obtained, a new node is added to the graph. The arc going from the initial node to the new node is labeled (l, e). The procedure is iterated until all basis markings have been considered. Note that, our approach always requires to enumerate a state space that is a strict subset of the reachability space. However, as in general for diagnosis approaches, the combinatory explosion cannot be avoided.

Example 7.3. Let us consider the PN in Figure 1, where $T_o = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7\}$, $T_u = \{\varepsilon_8, \varepsilon_9, \varepsilon_{10}, \varepsilon_{11}, \varepsilon_{12}, \varepsilon_{13}\}$, $T_f^1 = \{\varepsilon_{11}\}$ and $T_f^2 = \{\varepsilon_{12}\}$. The labeling function is defined as follows: $\mathcal{L}(t_1) = a$, $\mathcal{L}(t_2) = \mathcal{L}(t_3) = b$, $\mathcal{L}(t_4) = \mathcal{L}(t_5) = c$, $\mathcal{L}(t_6) = \mathcal{L}(t_7) = d$.

The BRG is shown in Figure 2. The notation used in in this figure is detailed in Tables 1 and 2. Each node contains a different basis marking and a binary row vector of dimension two, being two the number of fault classes. As an example, the binary vector $[0\ 0]$ is associated to M_0 because $\mathcal{T}(M_0)$ is not feasible for both fault classes. From node M_0 to node M_1 there is one arc labeled a and with the null vector as minimal explanation. The node containing the basis marking M_2 has binary vector $[0\ 1]$, because $\mathcal{T}(M_2)$ is feasible only for T_f^2. Node $(M_2, [0\ 1])$ has two output arcs both labeled with d and both directed to node $(M_1, [0\ 0])$ with two different minimal explanations $\vec{0}$ and e_1, respectively, plus another output arc $(b, \vec{0})$ directed to node $(M_4, [1\ 1])$. ■

The following algorithm summarizes the main steps of the on-line diagnosis carried out by looking at the BRG.

Algorithm 7.4 (Diagnosis using the BRG).

1. Let $w = \varepsilon$.
2. Let $\mathcal{M}(w) = \{(M_0, \vec{0})\}$.
3. Wait until a new observable transition fires.
Let l be the observed event.
4. Let $w' = w$ and $w = w'l$.
5. Let $\mathcal{M}(w) = \emptyset$, **[Computation of $\mathcal{M}(w)$]**
6. For all nodes containing M' : $(M', y') \in \mathcal{M}(w')$, do

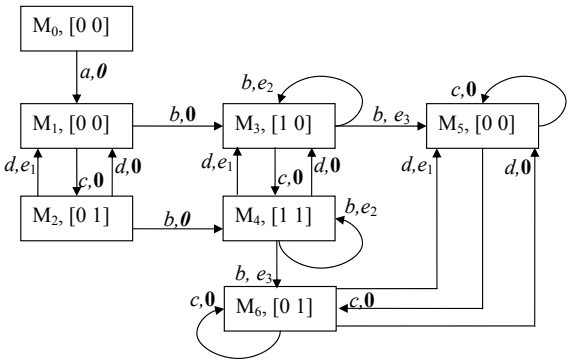


Figure 2: The BRG of the PN in Figure 1.

6.1. for all arcs exiting from the node with M', do
6.1.1. let M be the marking of the output node and e be the minimal e-vector on the edge from M' to M,
6.1.2. for all y' such that $(M', y') \in \mathcal{M}(w')$, do
6.1.2.1. let $y = y' + e$,
6.1.2.2. let $\mathcal{M}(w) = \mathcal{M}(w) \cup \{(M, y)\}$,
7. for all $i = 1, \ldots, r$, do
[Computation of the diagnosis state]
7.1. if $\forall\, (M, y) \in \mathcal{M}(w) \wedge \forall t_f \in T_f^i$ it is $y(t_f) = 0$, do
7.1.1. if $\forall\, (M, y) \in \mathcal{M}(w)$ it holds $x(i) = 0$, where x is the binary vector in node M, do
7.1.1.1. let $\Delta(w, T_f^i) = 0$,
7.1.2. else
7.1.2.1. let $\Delta(w, T_f^i) = 1$,
7.2. if $\exists\, (M, y) \in \mathcal{M}(w)$ and $(M', y') \in \mathcal{M}(w)$ s.t.:
(i) $\exists t_f \in T_f^i$ such that $y(t_f) > 0$,
(ii) $\forall t_f \in T_f^i$, $y'(t_f) = 0$, do
7.2.1. let $\Delta(w, T_f^i) = 2$,
7.3. if $\forall\, (M, y) \in \mathcal{M}(w)\ \exists t_f \in T_f^i : y(t_f) > 0$, do
7.3.1. let $\Delta(w, T_f^i) = 3$.
8. Goto step 3.

■

Steps 1 to 6 of Algorithm 7.4 enables us to compute the set $\mathcal{M}(w)$. When no event is observed, namely $w = \varepsilon$, then $\mathcal{M}(w) = \{(M_0, \vec{0})\}$. Now, assume that a label l is observed. We include in the set $\mathcal{M}(l)$ all couples (M, y) such that an arc labeled l exits from the initial node and ends in a node containing the basis marking M. The corresponding value of y is equal to the e-vector in the arc going from M_0 to M, being $\vec{0}$ the j-vector relative to M_0. In general, if w' is the actual observation, and a new event labeled l fires, we consider all couples $(M', y') \in \mathcal{M}(w')$ and all nodes that can be reached from M' with an arc labeled l. Let M be the basis marking of the generic resulting node. We include in $\mathcal{M}(w) = \mathcal{M}(w't)$ all couples (M, y), where for any M, y is equal to the sum of y' plus the e-vector labeling the arc from M' to M.

Step 7 of Algorithm 7.4 computes the diagnosis

Table 1: The markings of the BRG in Figure 2.

M_0	[	1	0	0	0	0	0	0	0	0	0	]^T
M_1	[	0	1	0	0	0	0	0	1	0	0	0]^T
M_2	[	0	1	0	0	0	0	0	0	1	0	0]^T
M_3	[	0	0	1	0	0	0	0	1	0	0	0]^T
M_4	[	0	0	1	0	0	0	0	0	1	0	0]^T
M_5	[	0	0	0	0	0	0	1	1	0	0	0]^T
M_6	[	0	0	0	0	0	0	1	0	1	0	0]^T

Table 2: The e-vectors of the BRG in Figure 2.

	ε_8	ε_9	ε_{10}	ε_{11}	ε_{12}	ε_{13}
e_1	0	0	0	0	1	1
e_2	1	1	1	0	0	0
e_3	1	0	0	1	0	0

state. Let us consider the generic ith fault class. If $\forall(M,y) \in \mathcal{M}(w)$ and $\forall t_f \in T_f^i$ it holds $y(t_f) = 0$, we have to check the ith entry of all the binary row vectors associated to the basis markings M, such that $(M,y) \in \mathcal{M}(w)$. If these entries are all equal to 0, we set $\Delta(w,T_f^i) = 0$, otherwise we set $\Delta(w,T_f^i) = 1$. On the other hand, if there exists at least one pair $(M,y) \in \mathcal{M}(w)$ with $y(t_f) > 0$ for any $t_f \in T_f^i$, and there exists at least one pair $(M',y') \in \mathcal{M}(w)$ with $y(t_f) = 0$ for all $t_f \in T_f^i$, then $\Delta(w,T_f^i) = 2$. Finally, if for all pairs $(M,y) \in \mathcal{M}(w)\ y(t_f) > 0$ for any $t_f \in T_f^i$, then $\Delta(w,T_f^i) = 3$.

The following example shows how to perform diagnosis on-line simply looking at the BRG.

Example 7.5. Let us consider the PN in Figure 1 and its BRG in Figure 2. Let $w = \varepsilon$. By looking at the BRG we establish that $\Delta(\varepsilon,T_f^1) = \Delta(\varepsilon,T_f^2) = 0$ being both entries of the row vector associated to M_0 equal to 0.

Now, let us consider $w = ab$. In such a case $\mathcal{M}(w) = \{(M_3,\vec{0})\}$. It holds $\Delta(ab,T_f^1) = 1$ and $\Delta(ab,T_f^2) = 0$ being the row vector in the node equal to [1 0].

Finally, for $w = abbc$ it holds $\Delta(abbc,T_f^1) = 2$ and $\Delta(abbc,T_f^2) = 1$. In fact $\mathcal{M}(w) = \{(M_4,y_1),(M_5,y_2)\}$, where $y_1 = e_2$, $y_2 = e_2 + e_3$, and the row vectors associated to M_4 and M_5 are respectively [1 1] and [0 0]. ■

8 MATLAB TOOLBOX

Our group at the University of Cagliari has developed a MATLAB toolbox for PNs.

In this section we illustrate how it can be used for the diagnosis of labeled PNs. In particular, we consider the function that given a bounded labeled PN builds the basis reachability graph.

The input of the MATLAB function BRG.m are:

- the structure of the net, i.e., the matrices *Pre* and *Post*;
- the initial marking M_0;
- a cell array F that has as many rows as the number of fault classes, that contains in each row the fault transitions that belong to the corresponding fault class;
- a cell array L that has as many rows as the cardinality of the considered alphabet, that contains in each row the observable transitions having the same label;
- a cell array E that contains in each row a string of characters, each one corresponding to a different label in the considered alphabet. Obviously, the cell array E is ordered according to L.

The output of the MATLAB function BRG.m is a cell array T that univocally identifies the resulting BRG. It has as many rows as the number of nodes of the BRG. A different row is associated to each node and contains the following information:

- an identifier number of the node;
- a matrix whose rows are equal to the transpose of the basis markings associated to the node;
- a matrix with as many rows as the number of basis markings associated to the node and as many columns as the number of fault classes: the jth element in the ith row (corresponding to M_b^i) is equal to $x_i(T_f^j)$ evaluated at M_b^i. Thus, $x_i(T_f^j) = 0$ is $\mathcal{T}(M_b^i)$ is not feasible with respect to T_j^f, 1 otherwise;

- the transitions enabled at node;
- the identifier number of the nodes that are reached firing an enabled transition and the corresponding j-vector.

9 NUMERICAL SIMULATIONS

Let us consider the Petri net in Figure 3 (Lai et al., 2008), where thick transitions represent observable event and thin transitions represent unobservable events. It models a family of manufacturing systems characterized by three parameters: n, m and k.

— n is the number of production lines.

— m is the number of units of the final product that can be simultaneously produced. Each unit of product is composed of n parts.

— k is the number of operations that each part must undergo in each line.

To obtain one unit of final product n orders are sent, one to each line; this is represented by observable event t_s. Each line will produce a part (all parts are identical) and put it in its final buffer. An assembly station will take one part from each buffer (observable event t_e) to produce the final product.

The part in line i ($i = 1,\ldots,n$) undergoes a series of k operations, represented by unobservable events $\varepsilon_{i,1},\varepsilon_{i,2},\cdots,\varepsilon_{i,k}$.

After this series of operations two events are possible: either the part is regularly put in the final buffer of the line, or a fault may occur.

— Putting the part in the final buffer of line 1 corresponds to unobservable event $\varepsilon_{1,k+1}$, while putting the part in the final buffer of line i ($i = 2,\ldots,n$) corresponds to observable event $t_{i,k+1}$.

— There are $n-1$ faults, represented by unobservable events f_i ($i = 1,\ldots,n-1$). Fault f_i moves a part from line i to line $i+1$. Note that on line i ($i = 1,\ldots,n-1$) the fault may only occur when the part has finished processing and is ready to be put in its final buffer; the part goes to the same processing stage in line $i+1$.

In this section we present the results of the computation of the BRG for several numerical simulations. Results obtained for different values of n, k and m are summarized in Tables 3, 4 and 5.

Note that for the sake of simplicity we assumed that all faults belong to the same class.

In these tables we also detail the cardinality of the reachability set R. This is an extremely important parameter to appreciate the advantage of using basis markings. The value of $|R|$ has been computed using a function we developed in MATLAB. For completeness we also reported the time necessary to compute it.

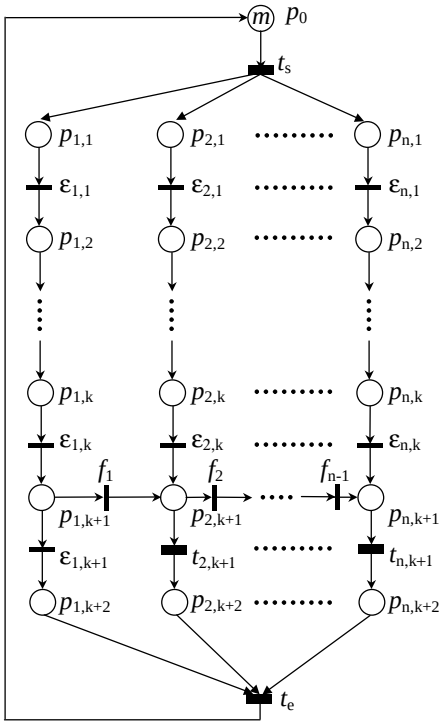


Figure 3: A manufacturing system.

Let us observe that some boxes of the above tables contain the non numerical values o:t: (out of time), that denotes that the corresponding value has not been computed within 6 hours.

All simulations have been run on a PC Athlon 64, 4000+ processor.

— Columns 1 and 2 show the values of n and k.

— Column 3 shows the number of nodes $|R|$ of the reachability graph.

— Column 4 shows the time t_R in seconds we spent to compute the reachability graph.

— Column 4 shows the number of nodes $|BRG|$ of the BRG.

— Column 5 shows the time t_{BRG} in seconds we spent to compute the BRG using the function BRG.m.

Tables 3, 4 and 5 show that the time spent to compute the reachability graph highly increases with the dimension of the net, namely with n and k, and with the number of products m.

On the contrary, the time spent to compute the BRG is always reasonable even for high values of n, k and m.

Tables 3, 4 and 5 also show that the number of nodes of the BRG only depends on n and m, while it is invariant with respect to k. On the other hand, $|R|$ also highly increases with k.

Table 3: Numerical results in the case of $m = 1$.

n	k	$\|R\|$	t_R [sec]	$\|BRG\|$	t_{BRG} [sec]
2	1	15	0.031	5	0.062
2	2	24	0.031	5	0.062
2	3	35	0.047	5	0.062
2	4	48	0.062	5	0.07
2	5	63	0.078	5	0.07
2	6	80	0.094	5	0.07
3	1	80	0.094	17	0.101
3	2	159	0.25	17	0.101
3	3	274	0.672	17	0.109
3	4	431	1.72	17	0.117
3	5	636	3.938	17	0.125
3	6	895	8.328	17	0.132
4	1	495	2.375	69	0.375
4	2	1200	16.969	69	0.43
4	3	2415	77.828	69	0.477
4	4	4320	272.53	69	0.531
4	5	7119	824.69	69	0.594
4	6	11040	2122.4	69	0.664
5	1	3295	155.81	305	4.345
5	2	9691	1615.7	305	4.765
5	3	22707	10288	305	5.25
5	4	*o.t.*	*o.t.*	305	5.75
5	5	*o.t.*	*o.t.*	305	6.897
5	6	*o.t.*	*o.t.*	305	7.894

Table 4: Numerical results in the case of $m = 2$.

n	k	$\|R\|$	t_R [sec]	$\|BRG\|$	t_{BRG} [sec]
2	1	96	0.11	17	0.086
2	2	237	0.469	17	0.094
2	3	496	2.078	17	0.1
3	1	1484	24.204	140	0.78
3	2	5949	486.39	140	0.844
3	3	18311	5320.9	140	0.906
4	1	28203	14006	1433	73.5
4	2	*o.t.*	*o.t.*	1433	76.5
4	3	*o.t.*	*o.t.*	1433	76.5

For the considered Petri net, on the basis of the above simulations, we can conclude that the diagnosis approach here presented is suitable from a computational point of view. In fact, thanks to the basis markings the reachability space can be described in a compact manner.

Table 5: Numerical results in the case of $m = 3$.

n	k	$\|R\|$	t_R [sec]	$\|BRG\|$	t_{BRG} [sec]
2	1	377	1.203	39	0.145
2	2	1293	17.203	39	0.145
3	1	12048	2113.9	553	8.219
3	2	*o.t.*	*o.t.*	553	9.016
4	1	*o.t.*	*o.t.*	9835	4095.06
4	2	*o.t.*	*o.t.*	9835	4095.06

10 CONCLUSIONS AND FUTURE WORK

This paper presents a diagnosis approach for labeled PNs using basis markings. This enables us to avoid an exhaustive enumeration of the reachability set. This approach applies to all bounded and unbounded Petri net systems whose unobservable subnet is acyclic. However, if we consider bounded net systems the most burdensome part of the procedure may be moved off-line computing the Basis Reachability Graph. Finally, we have presented a tool for the diagnosis of labeled bounded PNs and we have shown the simulation results using as diagnosis benchmark a family of manufacturing systems.

We have also studied the problem of diagnosability of bounded and unbounded PNs giving for both cases necessary and sufficient conditions for diagnosability. These results are not reported here, but they have been already submitted to an international conference.

Our future work will be that of studying the diagnosis problem for distributed systems investigating the possibility of extending the approach here presented to this case.

ACKNOWLEDGEMENTS

We thank Marco Pocci, a Master student of Electronic Engineering at the University of Cagliari, for his help in the development of the MATLAB tool for the construction of the BRG for labeled PNs.

REFERENCES

Basile, F., Chiacchio, P., and Tommasi, G. D. (2008). An efficient approach for online diagnosis of discrete event systems. *IEEE Trans. on Automatic Control*. in press.

Benveniste, A., Fabre, E., Haar, S., and Jard, C. (2003). Diagnosis of asynchronous discrete event systems: A

net unfolding approach. *IEEE Trans. on Automatic Control*, 48(5):714–727.

Boel, R. and Jiroveanu, G. (2004). Distributed contextual diagnosis for very large systems. In *Proc. IFAC WODES'04: 7th Work. on Discrete Event Systems*, pages 343–348.

Boel, R. and van Schuppen, J. (2002). Decentralized failure diagnosis for discrete-event systems with costly communication between diagnosers. In *Proc. WODES'02: 6th Work. on Discrete Event Systems*, pages 175–181.

Cabasino, M., Giua, A., and Seatzu, C. (2008). Fault detection for discrete event systems using Petri nets with unobservable transitions. *Automatica*. Preliminary accepted.

Cabasino, M., Giua, A., and Seatzu, C. (2009). Diagnosis of discrete event systems using labeled Petri nets. In *Proc. 2nd IFAC Workshop on Dependable Control of Discrete Systems (Bari, Italy)*.

Chung, S. (2005). Diagnosing pn-based models with partial observable transitions. *International Journal of Computer Integrated Manufacturing*, 12 (2):158–169.

Corona, D., Giua, A., and Seatzu, C. (2004). Marking estimation of Petri nets with silent transitions. In *Proc. IEEE 43rd Int. Conf. on Decision and Control (Atlantis, The Bahamas)*.

Debouk, R., Lafortune, S., and Teneketzis, D. (2000). Coordinated decentralized protocols for failure diagnosis of discrete-event systems. *Discrete Events Dynamical Systems*, 10(1):33–86.

Dotoli, M., Fanti, M., and Mangini, A. (2008). Fault detection of discrete event systems using Petri nets and integer linear programming. In *Proc. of 17th IFAC World Congress*, Seoul, Korea.

Genc, S. and Lafortune, S. (2007). Distributed diagnosis of place-bordered Petri nets. *IEEE Trans. on Automation Science and Engineering*, 4(2):206–219.

Ghazel, M., Togueni, A., and Bigang, M. (2005). A monitoring approach for discrete events systems based on a time Petri net model. In *Proc. of 16th IFAC World Congress*, Prague, Czech Republic.

Giua, A. and Seatzu, C. (2005). Fault detection for discrete event systems using Petri nets with unobservable transitions. In *Proc. 44th IEEE Conf. on Decision and Control*, pages 6323–6328.

Hadjicostis, C. and Veghese, G. (1999). Monitoring discrete event systems using Petri net embeddings. *Lecture Notes in Computer Science*, 1639:188–207.

Jiroveanu, G. and Boel, R. (2004). Contextual analysis of Petri nets for distributed applications. In *16th Int. Symp. on Mathematical Theory of Networks and Systems (Leuven, Belgium)*.

Lai, S., Nessi, D., Cabasino, M., Giua, A., and Seatzu, C. (2008). A comparison between two diagnostic tools based on automata and Petri nets. In *Proc. IFAC WODES'08: 9th Work. on Discrete Event Systems*, pages 144–149.

Lefebvre, D. and Delherm, C. (2007). Diagnosis of DES with Petri net models. *IEEE Trans. on Automation Science and Engineering*, 4(1):114–118.

Lin, F. (1994). Diagnosability of discrete event systems and its applications. *Discrete Event Dynamic Systems*, 4(2):197–212.

Lin, F., Markee, J., , and Rado, B. (1993). Design and test of mixed signal circuits: a discrete event approach. In *Proc. 32rd IEEE Conf. on Decision and Control*, pages 246–251.

Martinez, J. and Silva, M. (1982). A simple and fast algorithm to obtain all invariants of a generalized Petri net. In *Informatik-Fachberichte 52: Application and Theory of Petri Nets.*, pages 301–310. Springer-Verlag.

Murata, T. (1989). Petri nets: properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580.

Prock, J. (1991). A new tecnique for fault detection using Petri nets. *Automatica*, 27(2):239–245.

Ruiz-Beltràn, A. R.-T. E., Rivera-Rangel, I., and Lopez-Mellado, E. (2007). Online fault diagnosis of discrete event systems. A Petri net-based approach. *IEEE Trans. on Automation Science and Engineering*, 4(1):31–39.

Sampath, M., Lafortune, S., and Teneketzis, D. (1998). Active diagnosis of discrete-event systems. *IEEE Trans. on Automatic Control*, 43(7):908–929.

Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., and Teneketzis, D. (1995). Diagnosability of discrete-event systems. *IEEE Trans. on Automatic Control*, 40 (9):1555–1575.

Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., and Teneketzis, D. (1996). Failure diagnosis using discrete-event models. *IEEE Trans. Control Systems Technology*, 4(2):105–124.

Sreenivas, V. and Jafari, M. (1993). Fault detection and monitoring using time Petri nets. *IEEE Trans. Systems, Man and Cybernetics*, 23(4):1155–1162.

Ushio, T., Onishi, L., and Okuda, K. (1998). Fault detection based on Petri net models with faulty behaviors. In *Proc. SMC'98: IEEE Int. Conf. on Systems, Man, and Cybernetics (San Diego, CA, USA)*, pages 113–118.

Wen, Y. and Jeng, M. (2005). Diagnosability analysis based on T-invariants of Petri nets. In *Networking, Sensing and Control, 2005. Proceedings, 2005 IEEE.*, pages 371– 376.

Wen, Y., Li, C., and Jeng, M. (2005). A polynomial algorithm for checking diagnosability of Petri nets. In *Proc. SMC'05: IEEE Int. Conf. on Systems, Man, and Cybernetics*, pages 2542– 2547.

Wu, Y. and Hadjicostis, C. (2005). Algebraic approaches for fault identification in discrete-event systems. *IEEE Trans. Robotics and Automation*, 50(12):2048–2053.

Zad, S. H., Kwong, R., and Wonham, W. (2003). Fault diagnosis in discrete-event systems: framework and model reduction. *IEEE Trans. on Automatic Control*, 48(7):1199–1212.

BRIEF BIOGRAPHY

Alessandro Giua is professor of Automatic Control at the Department of Electrical and Electronic Engineering of the University of Cagliari, Italy. He received the Laurea degree in electric engineering from the University of Cagliari, Italy in 1988, and the M.S. and Ph.D. degrees in computer and systems engineering from Rensselaer Polytechnic Institute, Troy, New York, in 1990 and 1992.

His research interests include discrete event systems, hybrid systems, networked control systems, automated manufacturing, Petri nets, control of mechanical systems, failure diagnosis. He has co-authored two textbooks on Automatic Control (in Italian) and over 150 technical papers.

Dr. Giua is a member of the editorial board of the journals: Discrete Event Dynamic Systems: Theory and Applications; IEEE Trans. on Control Systems Technology; Nonlinear Analysis: Hybrid Systems. He has served in the program committee of over 60 international conferences.

He is chair for Chapter Activities of the Member Activities Board of the IEEE Control Systems Society and chair of the IFAC Technical Committee 1.3 on Discrete Event and Hybrid Systems.

MEETING THE WORLD CHALLENGES
From Philosophy to Information Technology to Applications

Peter Simon Sapaty
Institute of Mathematical Machines and Systems, National Academy of Sciences
Glushkova Ave 42, 03187 Kiev, Ukraine
sapaty@immsp.kiev.ua

Keywords: World crises, Atomism, holism, Gestalt theory, System integrity, Waves, Distributed scenario language, Networked interpretation, Unmanned systems, Task level, Behavioral level, Crisis management, Electronic warfare, Directed energy systems.

Abstract: We have been witnessing numerous world crises and disasters—from ecological to military to economic, with global world dynamics likely to be increasing this century further. The paper highlights known holistic and gestalt principles mainly used for a single brain, extending them to any distributed systems which may need high integrity and performance in reaction to unpredictable situations. A higher organizational layer is proposed enabling any distributed resources and systems to behave as an organism having global "consciousness" and pursuing global goals. This "over-operability" layer is established by implanting into key system points the same copy of a universal intelligent module, which can communicate with other such modules and interpret collectively global mission scenarios presented in a special Distributed Scenario Language. The scenarios can be injected from any module, and then self-replicate, self-modify, and self-spread throughout the system to be managed, tasking components, activating distributed resources, and establishing runtime infrastructures supporting system's integrity. Numerous existing and prospective applications are outlined and discussed, confirming paradigm's usefulness for solving hot world problems.

1 INTRODUCTION

To understand mental state of a handicapped person, problems of economy and ecology, or how to win on a battlefield, we must consider the system as a whole -- not just as a collection and interaction of parts. The situation may complicate dramatically if the system is dynamic and open, spreads over large territories, comprises unsafe or varying components, and cannot be observed in its entirety from a single point. Numerous world crises we have been witnessing at the beginning of this century, including the current economic one, may have emerged, first of all, due to our inability of seeing and managing complex systems as a whole.

To withstand the unwanted events and their consequences (ideally: predict and prevent them) we need effective worldwide integration of numerous efforts and often dissimilar and scattered resources and systems. Just establishing advanced communications between parts of the distributed systems and providing the possibility of sharing local and global information from any point, often called "interoperability", is becoming insufficient (even insecure and harmful) for solving urgent problems in dynamic environments, in real time and ahead of it.

We may need the whole distributed system to behave as an integral organism, with parts not so interoperating but rather complementing each other and representing altogether an integral whole pursuing global goals and having a sort of global awareness and consciousness. This whole should be essentially more than the sum of its parts, with the latter having sense, possibly even existence, in the context of this whole, rather than vice versa.

This paper develops further the over-operability principle researched in Sapaty, 1993, 1999, 2002, 2005 and other works (the term "over-operability" coined in Sapaty, 2002), which can establish intelligent dominant layer over distributed resources and systems, and help solve urgent world problems in a parallel, distributed, and dynamic way.

The rest of this paper compares the dominant atomistic approach in system design, implementation and management with holistic and gestalt principles, and describes a novel ideology and technology for integral solutions in distributed

worlds, which can avoid many traditional management routines in solving global problems, with its numerous practical applications outlined and discussed.

2 ATOMISM, HOLISM, GESTALT

We used to exercise predominantly atomistic, parts-to-whole philosophy of the system design, comprehension and implementation, which extends even to the organization of management facilities themselves -- as a collection of interacting parts, or *agents*. (This philosophy actually being the same as a century ago.)

Originally a system or campaign idea and the functionality needed emerge in a very general form (in a single human mind or in a close collective of such minds). Then this general idea (shown symbolically in Fig. 1*a*) is partitioned into individual chunks, or "atoms", each detailed and studied further (Fig. 1*b*). This logical partitioning already causes swelling of the problem complexity (as indicated in Fig. 1*b*).

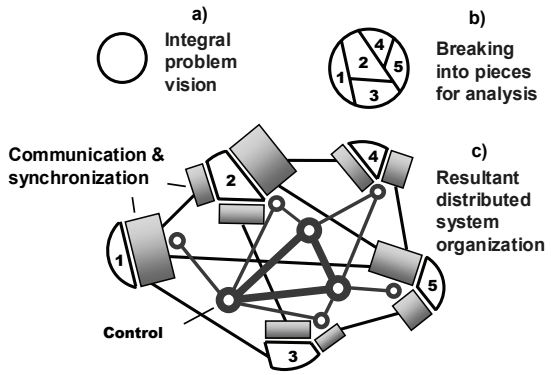


Figure 1: System overhead under atomistic organization.

The next step is materialization of the defined parts and their distribution in physical or virtual space. To make these parts work or behave together within the original idea of Fig. 1*a*, we may need a good deal of their communication and synchronization, also sophisticated control infrastructures, as depicted in Fig. 1*c*. This overhead may be considerable, outweighing and shadowing the original project definition.

The main problem is that the initial idea (Fig. 1*a*) and even its second stage (Fig. 1*b*) are usually non formalized, remaining in the minds of creators only, and the *real system description and implementation start from the already partitioned-interlinked stage, with its huge overhead* (as Fig. 1*c*).

This parts-to-whole approach also dominates in the controversial "society of mind" theory (Minsky, 1988), which is trying to explain even human thinking from the atomistic positions.

Holism (see, for example, Smuts, 2007) has quite an opposite vision of systems:

- Holism as an idea or philosophical concept is diametrically opposed to atomism.
- Where the atomist believes that any whole can be broken down or analyzed into its separate parts and the relationships between them, the holist maintains that the whole is primary and often greater than the sum of its parts.
- The atomist divides things up in order to know them better; the holist looks at things or systems in aggregate.

Gestalt theory (Koffka, 1913; Wertheimer, 1922) is based on the holistic principles too:

- For the gestaltists, "Gestalten" are not the sums of aggregated contents erected subjectively upon primarily given pieces.
- Instead, we are dealing with wholes and whole–processes possessed of inner intrinsic laws.
- *Elements* are determined as parts by the intrinsic conditions of their wholes and are to be understood *as parts* relative to such wholes."

Although gestalt psychology and theory was a general approach, most of the work on gestalt was done in the area of perception. *In our research, we are trying to use the holistic and gestalt principles for the organization of distributed systems with highest possible integrity and performance* (see Sapaty, 2009).

3 WAVES, FIELDS, SCENARIOS

We describe here a novel organizational philosophy and model, based on the idea of spreading *interdependent parallel waves* (as shown in Fig. 2), as an alternative to the dominant atomistic approach briefed above, also under the influence of mentioned holistic and gestalt ideas.

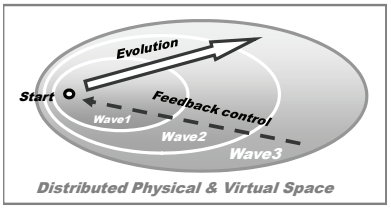


Figure 2: Grasping the entirety with spatial waves.

It allows us for an integral, parallel, and seamless navigation and coverage of virtual, physical or combined spaces where the solutions need to be found. Atomism emerges on the automatic implementation level only, which allows us to get high-level formal semantic definitions of systems and global operations in them, while omitting numerous organizational details (shown in Fig. 1*c*) and concentrating on global goals and overall performance instead.

An automatic materialization of this approach is carried out by the network of universal intelligent modules (U), embedded into important system points, which collectively interpret integral mission scenarios expressed in the waves formalism, which can start from any U, subsequently covering the distributed system at runtime.

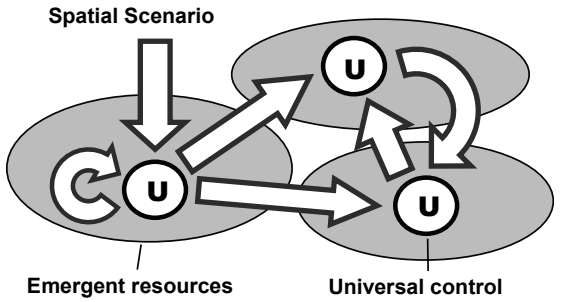


Figure 3: Self-spreading mission scenarios.

The wavelike scenarios are usually very compact and can be created and modified on the fly. They can cooperate or compete with each other in the distributed networked space as overlapping fields of parallel solutions.

Spreading waves can create knowledge infrastructures arbitrarily distributed between system components (robots, sensors, humans). These, subsequently or simultaneously navigated by same or other waves, can effectively support distributed databases, command and control, situation awareness, and autonomous decisions.

This paradigm is much in line with the existing abundant evidence that certain aspects of cognition, morals, needs, object relations, motor skills, and language acquisition proceed in developmental stages. These stages appear to be fluid, flowing, overlapping waves (Wilber, 2009), where also:

- Each stage has a holistic pattern that blends all of its elements into a structured whole;
- These patterns unfold in a relational sequence, with each senior wave transcending but including its juniors.

Our approach is also consistent with the ideas of self-actualization and person-centered approach (Rogers, 1978; Kriz, 2008), where the self is considered as an organized, consistent, conceptual gestalt exhibiting active forward thrust -- against tension reduction, equilibrium, or homeostasis (as in Freud, 2007, and others). In our case, instead of a single person we have the whole distributed system with high integrity and "active global thrust" behavior.

4 THE SCENARIO LANGUAGE

Distributed Scenario Language, or DSL (and its previous versions, WAVE including, as in Sapaty, 1999, 2005) reflects the waves model proposed, and allows us to directly express semantics of problems to be solved in distributed worlds, also the needed global system behavior in a non-atomistic manner. DSL operates with:

- *Virtual World (VW)*, which is discrete and consists of nodes and links connecting these nodes.
- Continuous *Physical World (PW)*, any point in which may be accessed by physical coordinates (taking into account certain precision).
- *Virtual-Physical World (VPW)*, which is an extension of VW where nodes additionally associate with certain coordinates in PW.

It also has the following key features:

- A DSL scenario develops as a transition between sets of progress points (or *props*) in the form of parallel *waves*.
- Starting from a prop, an action may result in one or more props (the resultant set of props may include the starting prop too).
- Each prop has a resulting *value* (which can be multiple) and a resulting *state* (being one of the four: *thru*, *done*, *fail*, and *abort*).
- Different actions may evolve independently or interdependently from the *same* prop, contributing to (and forming altogether) the resultant set of props.
- Actions may also *spatially succeed each other*, with new ones applied in parallel from all the props reached by preceding actions.
- Elementary operations can directly use local or remote values of props obtained from other actions (or even from the whole scenarios).
- Elementary operations can result either in open values that can be directly used as *operands* by other operations in an expression, or by the *next*

operations in a sequence. They can also be directly assigned to *local or remote variables* (for the latter case, an access to these variables may invoke scenarios of any complexity).

- Any prop can associate with a *node* in VW or a *position* in PW, or *both* -- when dealing with VPW.
- Any number of props can be simultaneously linked with the same points of the worlds.
- Staying with world points (virtual, physical, or combined) it is possible to *directly access* and update local data in them.
- Moving in physical, virtual or combined worlds, with their possible modification or even creation from scratch, are as routine operations as, say, arithmetic or logical operations of traditional programming languages.
- DSL can also be used as a usual universal programming language (like C, Java, or FORTRAN).

DSL has a recursive syntax, which on top level is as follows:

wave → ***phenomenon*** | ***rule*** **(** { ***wave*** **,** **}**)
phenomenon → ***constant*** | ***variable*** | ***special***
constant → ***information*** | ***matter*** | ***combined***
variable → ***heritable*** | ***frontal*** | ***environmental*** | ***nodal***
rule → ***movement*** | ***creation*** | ***elimination*** | ***echoing*** | ***fusion*** | ***verification*** | ***assignment*** | ***advancing*** | ***branching*** | ***transference*** | ***timing*** | ***granting***

Elementary programming examples in DSL are shown in Fig. 4 for: a) assignment of a sum of values to a variable; b) parallel movement into two physical locations; c) creation of a node in a virtual space, and d) extension of the latter with a new link and node.

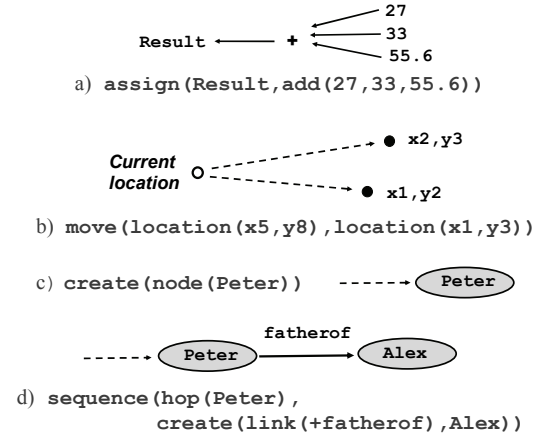


Figure 4: Elementary examples in DSL.

Traditional abbreviations of operations and delimiters can also be used, as in many further examples throughout this text, to simplify and shorten DSL programs, remaining however within the general recursive syntactic structure shown above.

5 COMPOSITION OF WAVES

The language allows for an integral parallel navigation of distributed worlds in a controlled *depth and breadth mode*, with any combinations of the two. We will highlight here key possibilities of doing this by composition of DSL scenarios, or waves.

5.1 Single Wave Features

Single wave (let it be *W1*) development features are shown in Fig. 5. Starting from a prop, which may be associated with a point in the world, the related scenario evolves, grasps, and covers certain region in it, performing any operations needed in the distributed space.

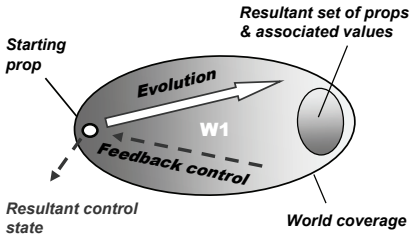


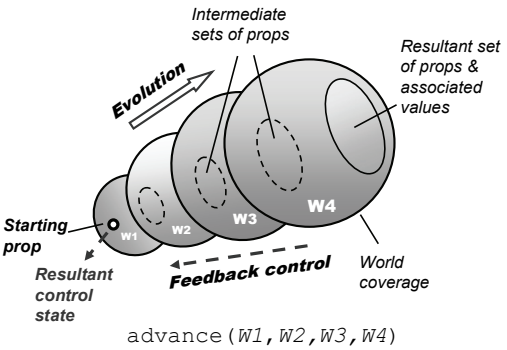
Figure 5: Single wave features.

The result of this spatial evolution may be multiple, and may lie in a (final) sub-region of the region covered, being represented by a set of resultant props (each linked to world points) and associated with them values. After termination of the wave, its resultant control state (which, in a parallel feedback process, merges termination states throughout the region covered) is available in the starting prop, and may be taken into account for decisions at higher levels. Also, if requested from higher levels, the values associated with the resultant props (which may be remote) can be lifted, spatially raked, and returned to the starting prop for a further processing.

5.2 Advancing in Space

The depth mode development of waves is shown in Fig. 6. For this type of composition, each subsequent

wave is applied in parallel from all props in space reached by the previous wave, with the resultant set of props (and associated values) on the whole group being the one of the last applied wave (i.e. *W4* in the figure).



`advance(W1,W2,W3,W4)`

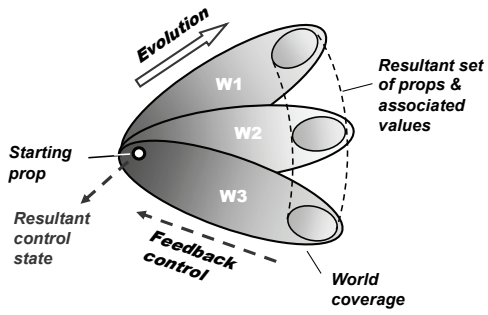
Figure 6: Depth mode composition of waves.

This spatial advancement of waves returns the resultant control state which is available at the starting prop, and the values of the resultant set of props can also be echoed to the starting prop if requested. Examples of other advancing rules:

- `advance synchronized` – the one where any new wave is applied only after all invocations of the previous wave have been terminated;
- `repeat` – where the same wave is applied repeatedly from all props reached by its previous invocation;
- `repeat synchronized` – where in the repeated invocation of a wave each new invocation starts only after full completion of the previous one.

5.3 Branching in Space

The branching breadth mode composition of waves is shown in Fig. 7, where all waves in the group are evolving from the same starting prop, and each wave, with its own resultant set of props and associated values, contributes to the final result.



`sequence(W1,W2,W3)`

Figure 7: Breadth mode composition of waves.

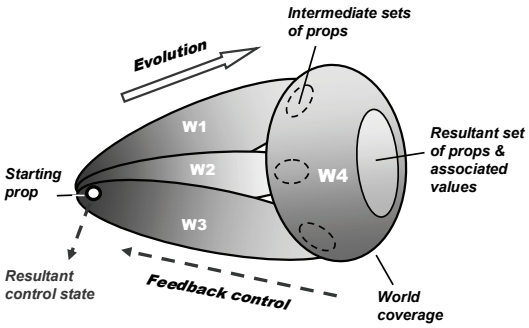
The merge of results from different waves depends on the branching rule used, with their repertoire (besides the `sequence` in Fig. 7) including:

`if, while, parallel, or, parallel or, and, parallel and, cycle, loop, and sling.`

(More details on these and other rules can be found, say, from Sapaty, 1999, 2005.)

5.4 Combined Branching-Advancing

Any combination of advancing and branching modes in a distributed space can be expressed and implemented in DSL (as shown in Fig. 8).



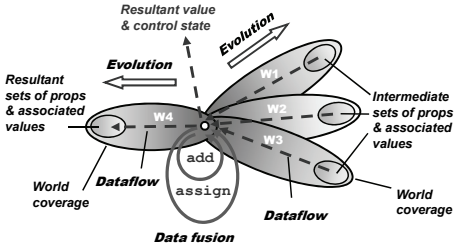
`advance(sequence(W1,W2,W3),W4)`

Figure 8: Breadth–depth composition mode.

These combinations, when embraced by the existing variety of composition rules, can provide any imaginable and even so far unimaginable spatial algorithms that can solve distributed problems in highly integral and compact ways, without explicit descending to the traditional atomistic level shifted to the automatic implementation only.

5.5 Operations on Remote Values

Due to fully recursive organization of DSL, it is possible to program in it arbitrary complex expressions directly operating not only on local but also arbitrarily remote values, where any programs (scenarios) can happen to be operands of any operations (expressed by rules). This gives an enormous expressive power and compactness to complex spatial scenarios evolving in distributed environments. An example of such compact expression of spatial operations on remote values and variables is shown in Fig. 9.




```
assign(W4,add(W1,W2,W3))
```

Figure 9: Direct operations on remote values.

6 DISTRIBUTED INTERPRETER

DSL interpreter, as from the previous language version called WAVE (Sapaty, 1993, 1999, 2005), has been prototyped in different countries on various platforms. Its public domain version (financed in the past by Siemens/Nixdorf) is being used for applications like intelligent network management or simulation of distributed dynamic systems. The DSL interpreter basics include:

- It consists of a *number of specialized modules* working in parallel and handling and sharing specific data structures, which are supporting persistent virtual worlds and temporary hierarchical control mechanisms.
- The whole network of the interpreters can be *mobile and open*, changing at runtime the number of nodes and communication structure between them.
- The heart of the distributed interpreter is its *spatial track system* enabling hierarchical command and control and remote data and code access, with high integrity of emerging parallel and distributed solutions.

The DSL interpreter structure is shown in Fig. 10.

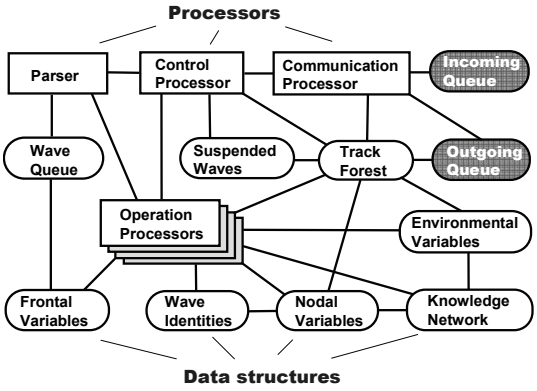


Figure 10: Structure of DSL interpreter.

It can be easily implemented in both software and hardware on any platforms, where the intelligent "wave chip" can be implanted into a great variety of devices, making them working together as an integral unit under the spatial DSL scenarios.

7 PROGRAMMING EXAMPLES

We will show here examples of solution in DSL of some important problems on networks and graphs in a fully distributed way, where each node may reside in a separate computer.

7.1 Shortest Paths

The solution for finding a path between two nodes by navigating the network with parallel waves is sown in Fig. 11, and the scenario that follows.

```
sequence(
 (direct # a; Ndist = 0; repeat(
  any #; Fdist += LINK;
  Ndist == nil, Ndist > Fdist;
  Ndist = Fdist; Npred = BACK))
 (direct # e; repeat(
  Fpath &= CONT; any # Npred);
  USER = Fpath))
```

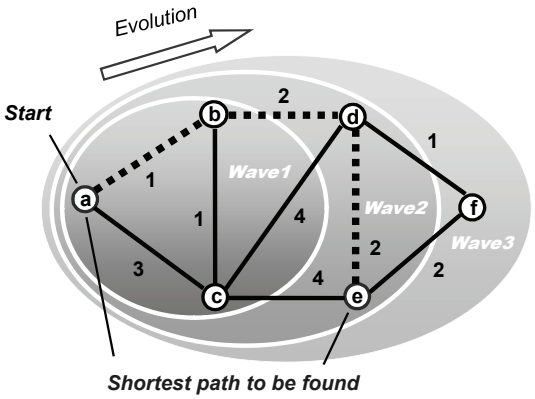


Figure 11: Finding shortest path with waves.

Many problems of optimization and control may be expressed as finding shortest paths in a distributed solution space.

7.2 Spatial Topology Analysis

DSL allows us to directly analyze and process distributed topologies in a parallel and extremely concise way.

7.2.1 Articulation Points

To find the weakest nodes in a network (called *articulation points*) which, when removed, split it into disjoint parts, as in Fig. 12 for node `d`, we need only the program that follows.

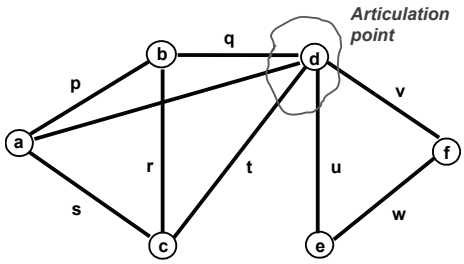


Figure 12: Articulation points.

```
direct # all; ID = CONT; Nm = 1;
and((random(all #);
 repeat(Nm ==; Nm = 1; all #)),
 (all #; Nm ==), USER = CONT)
```

Result: d.

7.2.2 Cliques

Cliques (or fully connected sub-graphs of a graph, as in Fig. 13), on the opposite, may be considered as strongest parts of a system. They can be found in parallel by the program that follows.

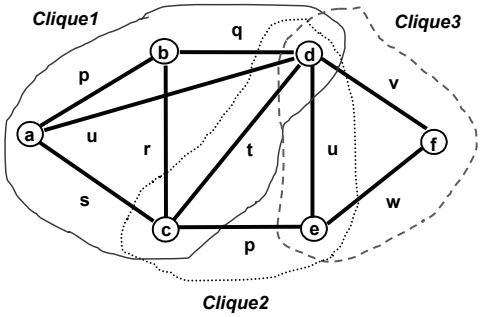


Figure 13: Cliques.

```
direct # all; Fclique = CONT;
repeat(all #; CONT !~ Fclique;
 and(andpar(any # Fclique; done !),
  or((BACK > CONT; done !),
   Fclique &= CONT))); USER = Fclique
```

Result: (a,b,c,d), (c,d,e), (d,e,f)

7.2.3 All Triangles

Any topological patterns can be found in any distributed network. For example, finding all triangles in a graph in Fig. 13 needs a simple code:

```
direct # all; Ftr = CONT;
2(all#; BACK > CONT; Ftr &= CONT);
 any # Ftr : 1; USER = Ftr
```

Result: (a,b,c), (b,c,d), (c,d,e), (d,e,f), (a,b,d), (a,c,d)

7.2.4 Network Creation

Any network can be created in a distributed space, and in parallel mode, by a very simple code too, as follows, as for the network in Fig. 13.

```
create(direct#a; p#b; q#d; u##a,(v#f;
w#e; u##d,(p#c; s##a, r##b, t##d)))
```

Arbitrary infrastructures can be created at runtime, on the fly, which can become active by putting certain procedures into their nodes and links. Any other existing models (incl. Petri nets, neural nets, contract nets, etc.) can also be implemented in a fully distributed and parallel way in DSL. Many related examples can be found in Sapaty, 1999.

8 COLLECTIVE ROBOTICS

Installing DSL interpreter into mobile robots (ground, aerial, or underwater) may allow us to organize any group solutions of complex problems in distributed physical spaces in a concise and effective way, shifting traditional management routines to automatic level. It is possible to express tasks and behaviors on different levels, as follows.

8.1 Task Level

Heterogeneous groups of mobile robots (as in Fig. 14) can be tasked at a highest possible level, just telling what they should do together, without detailing how, and what are the duties of every unit. An example task may be formulated as follows.

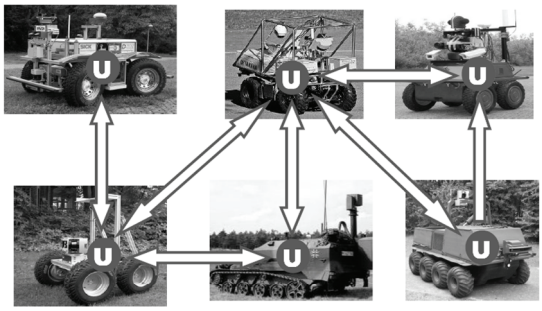


Figure 14: Grouping ground vehicles.

Go to physical locations of the disaster zone with coordinates (50.433, 30.633), (50.417, 30.490), and (50.467, 30.517). Evaluate damage in each location, find and transmit the maximum destruction value, together with exact coordinates of the corresponding location, to a management center.

The DSL program will be as follows:

```
transmit(maximum(
  move((50.433, 30.633),
       (50.417, 30.490),
       (50.467, 30.517));
  evaluate(destruction)& WHERE))
```

Details of automatic implementation of this scenario by different numbers of mobile robots are discussed in (Sapaty, 2009c).

8.2 Behavioral Level

After embedding DSL interpreters into robotic vehicles (like the aerial ones in Fig. 15), we can also provide any needed detailed collective behavior of them (at a lower than top task level, as before)—from *loose swarms* to a *strictly controlled integral unit* obeying external orders. Any mixture of different behaviors within the same scenario can be easily programmed too.

The following DSL scenario combines loose, random swarm movement in a distributed space with periodic finding/updating topologically central unit, and setting runtime hierarchical infrastructure between the units. The latter controls observation of distributed territory, collecting potential targets, distributing them between the vehicles, and then impacting potential targets by them individually. More on the implementation of this scenario can be found in Sapaty, 2008.

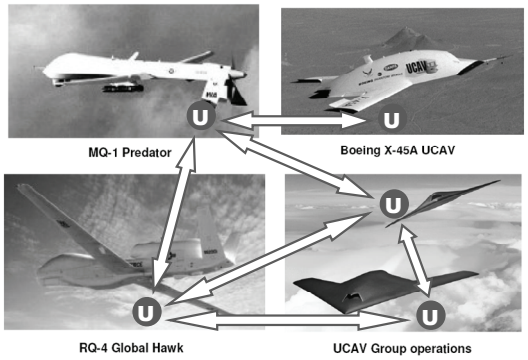


Figure 15: Grouping aerial vehicles.

```
(hop(allnodes); Range = 500;
 Limits = (dx(0,8), dy(-2,5));
 repeat(Shift = random(Limits);
  if(empty(hop(Shift, Range),
     move(Shift)))),
(repeat(hop(
  Faver =average(hop(allnodes);WHERE);
  min(hop(allnodes);
  distance(Aver, WHERE)& ADDRESS):2));
  stay(hop(nodes,all);rem(links,all);
  Frange = 20; repeat(
   linkup(+infra, firstcome, Frange));
  orpar(
   loop(nonempty(Fseen =
    repeat(free(detect(targets)),
    hoplinks(+ infra));
    repeat(
     free(select_move_shoot(Fseen),
     hoplinks(+ infra))),
   sleep(360)))
```

9 OTHER APPLICATIONS

9.1 Distributed Avionics

Distributed communicating DSL Interpreters, embedded into aircraft's key mechanisms (as in Fig. 16), can provide highest possible integrity of the aircraft that may continue to *function as a whole* even under *physical disintegration* -- which may help find critical runtime solutions saving lives and equipment (see also Sapaty, 2008a).

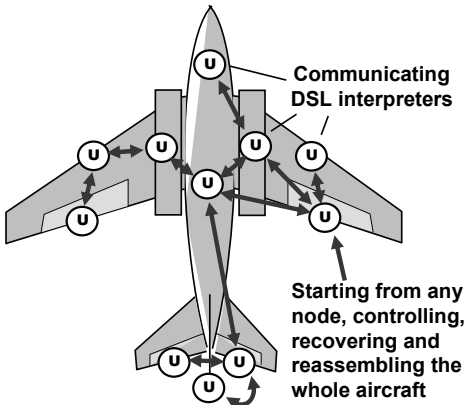


Figure 16: Distributed control infrastructure.

Collecting availability of aircraft's basic mechanisms, and establishing overall aircraft control from any available DSL interpreter, may be organized as follows:

```
Available =
 repeat(free(belong(CONT,
  (left_aileron, right_aileron,
   left_elevator,right_elevator,
   rudder, left_engine,right_engine,
   left_chassis, right_chassis, …));
```

```
   CONT), hop(firstcome, neighbors));
if(sufficient(Available),
   control(Available), set(alarm))
```

9.2 Objects Tracking

In a large distributed space, each embedded (or moving) sensor can handle only a limited part of space, so to keep the whole observation continuous, the mobile object seen should be handed over between neighboring sensors during its movement, along with the data accumulated on it (see also Sapaty, 1999, 2007, 2008).

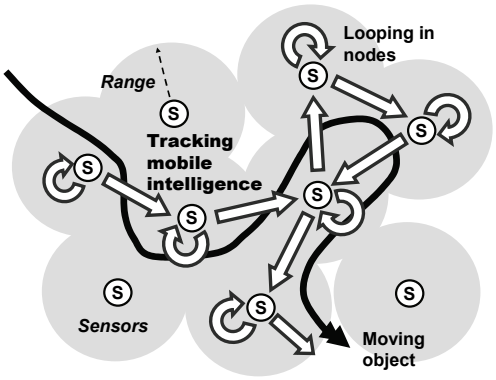


Figure 17: Tracking mobile objects.

The following program, starting in all sensors, *catches the object* it sees and *follows* it wherever it goes, if not observable from this point any more.

```
hop(allnodes); Fthr = 0.1;
Fobj = search(aerial);
visibility(Fobj) > Fthr; repeat(
 loop(visibility(Fobj) > Fthr);
 maxdest(hop(neighbors); (Seen =
 visibility(Fobj)) > Fthr; Seen))
```

9.3 Emergency Management

Embedded communicating DSL Interpreters can convert any post-disaster wreckage into a universal spatial machine capable of self-analysis and self-recovery under integral management scenarios (as in Sapaty, Sugisaka, Finkelstein, Delgado-Frias, Mirenkov, 2006; Sapaty, 2006). For example, all casualties counting program may be as follows (with its distributed operation shown in Fig. 18):

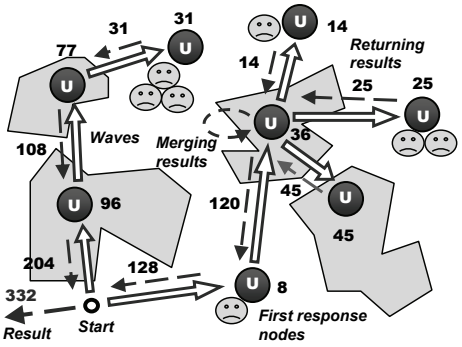


Figure 18: Counting all casualties.

```
Farea = disaster area definition;
output(sum(hop(Farea);
 repeat(free(count(casualties)),
 hop(alllinks, firstcome, Farea))))
```

Counting casualties in each region separately and organizing proportional relief delivery to each of them, may be expressed as follows:

```
Frea = disaster area definition;
split(collect(hop(Farea));
 repeat(done(count(casualties)&WHERE),
   hop(anylinks, firstcome, Farea))));
Fsupply = replicate("package", VAL:1);
move(VAL:2); distribute(Fsupply)
```

9.4 Directed Energy Systems

Directed energy systems and weapons are of rapidly growing importance in many areas, and especially in critical infrastructure protection, also on advanced battlefields (as shown in fig. 19). With the hardware equipment operating with the speed of light, traditional manned C2 is becoming a bottleneck for these advanced technical capabilities. With the technology offered, we may organize any runtime C2 infrastructures operating automatically, with the "speed of light" too, fitting the hardware capabilities and excluding men from the loop in time critical situations.

Figure 19: DEW in an advanced battlespace.

The following is an example of setting an automatic runtime C2 in a system with direct energy (DE) source, relay mirror (RM), and a target discovered, with an operational snapshot shown in Fig. 20.

```
sequence(
  parallel(
    (hop(DE); adjust(RM)),
    (hop(RM); adjust(DE, Target))),
  (hop(DE); activate(DE)))
```

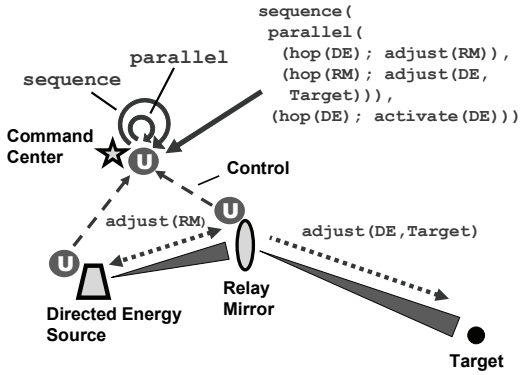


Figure 20: DE-RM-target operational snapshot.

There also exist advanced projects of global dominance with transference of directed energy, like the Boeing's Advanced Relay Mirror System (ARMS) concept. It plans to entail a constellation of as many as two dozen orbiting mirrors that would allow 24/7 coverage of every corner of the globe. When activated, this would enable a directed energy response to critical trouble spots anywhere.

We can use the distributed shortest path solution shown in section 7.1 for providing a runtime path in a worldwide distributed *dynamic* set of relay mirrors (as some of which may happen to be out of order) -- between the DE source and destination needed. This will enable optimal directed energy transfer, as shown in Fig. 21 (see also Sapaty, Morozov, Sugisaka, 2007).

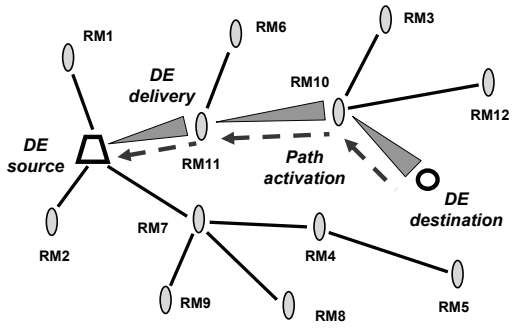


Figure 21: DE delivery via network of relay mirrors.

9.5 Electronic Warfare

Often the picture in Fig. 22 is shown as a typical example of electronic warfare. But this may rather be the last chance to survive from a missile attack. Involvement of many diverse and interlinked systems, especially for preventing and anticipating the attacks, which may be multiple and simultaneous, should be of paramount importance. All existing and being developed electronic support, attack, and protection measures have very limited scope and effect if used alone. But taken together they may provide a capability for fulfilling the objectives required. And the technology offered can readily organize this (as in Sapaty, 2007a, 2009a).

Figure 22: A Lockheed plane releasing decoy flares.

Instead of *physical flares* thrown from a plane in the final moments, we may throw, throughout the region in danger, which may be worldwide, the "*DSL scenario flares"* that can dynamically unite any available DE facilities and systems in an overwhelming electronic response to any threats.

9.6 Robotized Armies

Distributed robotized systems are of rapidly growing importance in defense (Singer, 2009, 2009a), where robotic swarming on asymmetric battlefields is becoming a major dimension of the new military doctrine for 21st century. But, as admitted by Singer, 2009, swarming, along with its simple rules of individual behavior and fully distributed nature, agility, and ubiquity, may also result in unpredictability of behavior for both sides of the conflict.

The approach briefed in this paper, also investigated in previous publications on this paradigm, is very much in line with these modern trends. Moreover, we are offering a unified solution that can harness loosely coupled swarms, always

guaranteeing their global-goal-driven behavior, where the watershed between loose swarming and strict hierarchical control may be situation dependent and changing over time (as programmed in Section 8.2).

These new doctrine trends will inevitably influence the role and sense of communications on battlefields, as with the planned drastic reduction of centralized C2 much more emphasis will be paid to intelligent tactical communications, where the scenario mobility in networked systems, offered by the approach proposed, may constitute an effective solution, with the key points (as in Sapaty, 2009b):

- Dramatic shift of global organization to intelligent tactical communications;
- Self-spreading and self-recovering mission scenarios and emergent command and control;
- Embedding intelligent protocol module into existing communication equipment;
- Situation-dependent watershed between global control and local communications.

In relation to the said above, different (including new) types of commands and control strategies for distributed robotized systems were investigated in DSL (Sapaty, Morozov, Sugisaka, Finkelstein, Lambert, 2008).

10 THE FIRST COMPUTERS

The approach offered may be compared with the invention of the first world computers (Rojas, 1997) and first high-level programming languages (Zuse, 1948/49). In our case, this computer may not only operate with data stored in a localized memory, but can cover, grasp, and manage any distributed system, the whole world including, and can work not only with information but with physical matter or objects too.

If compared with the Turing computational model, instead of the head moving through tape in performing calculations, we have a recursive formula that unwraps, replicates, covers and matches the distributed world in parallel, scanning it forth and back, bringing operations and data directly to the points of their consumption, automatically setting distributed command and control infrastructures, and organizing local and global behaviors needed.

The term "computer" first referred to the people who did scientific calculations by hand (Grier, 2005). In the end, they were rewarded by a new electronic machine that took the place and the name of those who were, once, the computers.

We can draw the following symbolic parallel with this. Despite the overwhelming automation of human activity (in both information and matter processing) the world as a whole may still be considered as remaining a *human machine*, as main decisions and global management still remain the human prerogative.

With the approach offered, we can effectively automate this top-level human supervision, actually converting the whole world into a *universal programmable machine* spatially executing global scenarios in DSL or a similar language. Despite certain science fiction flavor of this comparison, we can find numerous applications for such a global approach, some mentioned above, where top level decision automation could withstand unwanted events and save lives, and where timely human reaction may not be possible, even in principle.

11 CONCLUSIONS

We have developed and tested a novel system approach, which can describe what the system should do and how to behave on a higher level, while delegating traditional management details (like partitioning into components, their distribution, interaction and synchronization) to the effective automatic layer.

A DSL scenario is not a usual program -- it is rather a recursive active spatial pattern dynamically matching structures of distributed worlds. It has a hierarchical organization, which is grasping, by means of spreading parallel waves, the whole of the system to be comprehended and impacted.

The DSL scenarios can also create, in a parallel and fully distributed way, active distributed worlds, which become persistent and operate independently. They may spatially intervene into operation of these and other worlds and systems, changing their structures and behaviors in the way required, also self-recover from indiscriminate failures and damages, as well as repair and recover the systems managed.

Prospective applications of this work can also be linked with economy, ecology and weather prediction—by using the whole networked world as a spatial supercomputer, self-optimizing its performance. Also, for terrorism and piracy fight, where the powerful parallel ability of analyzing distributed systems and finding strong and weak patterns in them, as well as any structures (as shown in Section 7.2) may be the key to global solutions.

Crises may spark anywhere and anytime like, say, birds or swine flu or the current global economic disaster. We must be ready to react on them quickly and asymmetrically, withstanding and eradicating them -- in a "pandemic" way too, highly organized and intelligent, however.

We already have technical capabilities for this, as for example, the number of mobile phone owners in the world is approaching 3bn, and installing DSL interpreter in at least a fraction of them, can allow us to organize collective runtime (and ahead of it) response to any world events.

ACKNOWLEDGEMENTS

This work has been sponsored by the Alexander von Humboldt Foundation in Germany. Special thanks to Anatoly Morozov, Vitaly Klimenko, Yuri Korovitsky, Vladimir Voloboev, Eugene Bondarenko, and Anatoly Belyaev from the National Academy of Sciences of Ukraine for years of invaluable support and productive, often hot, discussions of these ideas. Recent chats with Juergen Kriz at the 16th GTA Convention in Osnabrueck, Germany, strengthened author's admiration of the gestalt psychology and theory, influencing general orientation of this paper.

REFERENCES

Freud, S., 1997. *General Psychological Theory*. Touchstone.

Grier, D. A., 2005. When Computers Were Human, Princeton University Press.

Koffka, K., 1913. Beiträge zur Psychologie der Gestalt. *von F. Kenkel. Zeits. f.Psychol*., 67.

Kriz, J., 2008. *Self-Actualization: Person-Centred Approach and Systems Theory*. PCCS Books.

Minsky, M., 1988. *The Society of Mind.* Simon and Schuster, New York.

Rojas, R., 1997. Konrad Zuse's Legacy: The Architecture of the Z1 and Z3. *IEEE Annals of the History of Computing*, Vol. 19, No. 2.

Rogers, C. R., 1978. *Carl Rogers on Personal Power: Inner Strength and Its Revolutionary Impact*. Trans-Atlantic Publications.

Sapaty P., 2009. Gestalt-Based Ideology and Technology for Spatial Control of Distributed Dynamic Systems, *Proc. International Gestalt Theory Congress, 16th Scientific Convention of the GTA*. University of Osnabrück, Germany.

Sapaty, P., 2009a. Distributed Capability for Battlespace Dominance. In *Electronic Warfare 2009 Conference & Exhibition*, Novotel London West Hotel & Conference Center, London.

Sapaty, P., 2009b. High-Level Communication Protocol for Dynamically Networked Battlefields. *Proc. International Conference Tactical Communications 2009 (Situational Awareness & Operational Effectiveness in the Last Tactical Mile)*. One Whitehall Place, Whitehall Suite & Reception, London, UK.

Sapaty, P. S., 2009c. Providing Spatial Integrity For Distributed Unmanned Systems". *Proc. 6th International Conference in Control, Automation and Robotics ICINCO 2009*. Milan, Italy.

Sapaty, P., 2008. Distributed Technology for Global Dominance. *Proc. of SPIE* -- Volume 6981, Defense Transformation and Net-Centric Systems 2008, Raja Suresh, Editor, 69810T.

Sapaty, P., 2008a. Grasping the Whole by Spatial Intelligence: A Higher Level for Distributed Avionics. *Proc. International Conference Military Avionics 2008*, Cafe Royal, London, UK.

Sapaty, P., Morozov, A., Finkelstein, R., Sugisaka, M., Lambert, D., 2008. A new concept of flexible organization for distributed robotized systems. *Artificial Life and Robotics*, Volume 12, Nos. 1-2/ March, Springer Japan.

Sapaty, P., 2007. Intelligent management of distributed sensor networks. In *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense VI*, edited by Edward M. Carapezza, *Proc. of SPIE*, Vol. 6538, 653812.

Sapaty, P., 2007a. Global Management of Distributed EW-Related Systems. *Proc. International Conference Electronic Warfare: Operations & Systems*. Thistle Selfridge, London, UK.

Sapaty, P., Morozov, A., Sugisaka, M., 2007. DEW in a Network Enabled Environment. *Proc. International Conference Directed Energy Weapons 2007*. Le Meridien Piccadilly, London, UK.

Sapaty, P., 2006. Crisis Management with Distributed Processing Technology. *International Transactions on Systems Science and Applications*. Vol. 1, no. 1.

Sapaty, P., Sugisaka, M., Finkelstein, R., Delgado-Frias, J., Mirenkov, N., 2006. Advanced IT Support of Crisis Relief Missions. *Journal of Emergency Management.* Vol.4, No.4, July/August.

Sapaty, P. S., 2005. *Ruling Distributed Dynamic Worlds*. John Wiley & Sons, New York.

Sapaty, P. S., 2002. Over-Operability in Distributed Simulation and Control. *The MSIAC's M&S Journal Online*. Winter Issue, Volume 4, No. 2, Alexandria, VA, USA.

Sapaty, P. S., 1999. *Mobile Processing in Distributed and*

Sapaty, P. S., 1993. A distributed processing system, *European Patent No. 0389655*. European Patent Office.

Singer, P. W., 2009. Wired for War. Robots and Military Doctrine. *JFQ / issue 52*, 1st quarter.

Singer, P. W., 2009a. *Wired for War: The Robotics Revolution and Conflict in the 21st Century*. Penguin.

Smuts, J. C., 2007. *Holism And Evolution*. Kessinger Publishing, LLC

Wertheimer, M., 1924. *Gestalt Theory*. Erlangen, Berlin.

Wilber, K. 2009. *Ken Wilber Online: Waves, Streams, States, and Self--A Summary of My Psychological Model (Or, Outline of An Integral Psychology)*. Shambhala Publications.

Zuse, K., 1948/49. Uber den Plankalk, als Mittel zur Formulierung schematisch kombinativer Aufgaben. In *Archiv Mathematik*, Band I.

BRIEF BIOGRAPHY

Dr Peter Simon Sapaty, chief research scientist and director of distributed simulation and control project at the Institute of Mathematical Machines and Systems, National Academy of Sciences of Ukraine, is with networked systems for more than four decades. A power network engineer on education, he created citywide heterogeneous computer networks from the end of the sixties, implemented a multiprocessor macro-pipeline supercomputer in the seventies-eighties, and since then used distributed computer networks for solving complex problems of most different natures—from distributed knowledge bases to intelligent network management to road traffic control to simulation of battlefields. He also worked in Germany, UK, Canada, and Japan as Alexander von Humboldt Foundation fellow, project leader, research professor, department head, and special invited professor; created and chaired a SIG on mobile cooperative technologies within Distributed Interactive Simulation project in the US. Peter invented a higher-level distributed networking technology used in different countries and resulted in a European Patent and two John Wiley books. His interests include coordination and simulation of large distributed dynamic systems under the holistic and gestalt principles, with application in advanced command and control, cooperative robotics, infrastructure protection, crisis management, and especially for finding asymmetric solutions in unpredictable and hostile environments.

INTELLIGENT CONTROL SYSTEMS AND OPTIMIZATION

FULL PAPERS

A TARGET TRACKING ALGORITHM BASED ON ADAPTIVE MULTIPLE FEATURE FUSION

Hongpeng Yin, Yi Chai
College of Automation, Chongqing University, Chongqing City, 400030, China
yinhongpeng@gmail.com, chaiyi@cqu.edu.cn

Simon X. Yang
School of Engineering, University of Guelph, Guelph, Ontario, N1G 2W1, Canada
syang@uoguelph.ca

David K. Y. Chiu
Dept. of Computing and Information Science, Univ. of Guelph, Guelph, ON, N1G 2W1, Canada
dchiu@cis.uoguelph.ca

Keywords: Target tracking, Feature fusion, Template update, Kernel-based tracking.

Abstract: This paper presents an online adaptive multiple feature fusion and template update mechanism for kernel-based target tracking. According to the discrimination between the object and background, measured by two-class variance ratio, the multiple features are combined by linear weighting to realize kernel-based tracking. An adaptive model-updating mechanism based on the likelihood of the features between successive frames is addressed to alleviate the mode drifts. In this paper, RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are employed to implement the scheme. Experiments on several video sequences show the effectiveness of the proposed method.

1 INTRODUCTION

Visual tracking is a common task in computer vision and play key roles in many scientific and engineering fields. Various applications ranging from video surveillance, human computer interaction, traffic monitoring to video analysis and understanding, all require the ability to track objects in a complex scene. Many powerful algorithms for target tracking have yielded two decades of vision research. Frame difference and adaptive background subtraction combined with simple data association techniques can effectively track in real-time for stationary cameras target tracking (Collins et al., 2001; Shalom and Fortmann, 1988; Stauffer and Grimson, 1995). Optical flow methods using the pattern of apparent motion of objects, surfaces and edges in a visual scene caused by the relative motion between the camera and scene. These methods can achieve the target tracking in the stationary cameras scene and the mobile cameras scene (Barron and Fleet, 1994; Tal and Bruckstein, 2008). Modern appearance-based methods using the likelihood between the tracked target appearance describe model and the reference target appearance describe model can achieve the target tracking without prior knowledge of scene structure or camera motion. Modern appearance-based methods include the use of flexible template models (Wang and Yaqi, 2008; Mattews et al., 2004) and kernel-based methods that track nonrigid objects used colour histograms (Comaniciu and Meer, 2002; Comaniciu et al., 2003; Li et al., 2008). Particle filter and Kalman filter are using to achieve more robust tracking of manoeuvring objects by introducing statistical models of object and camera motion (Comaniciu et al., 2003; Pan et al., 2008; Chang et al., 2008; Maggio et al., 2007).

The major difficulty in target tracking based on computer vision is the variation of the target appearance and its background. By using a stationary camera, the background in a long image sequence is dynamic. However, the performance of a tracker can be improved by using an adaptive tracking scheme and multiple features. The basic ideal is online adaptive selection of appropriate features for tracking. Recently, several adaptive tracking algorithms (Collins et al., 2005; Wang and Yaqi, 2008; Wei and Xiaolin,

2007; Dawei and Qingming, 2007; Zhaozhen et al., 2008)were proposed. Collins et al. proposed to online select discriminative tracking features from linear combination of RGB values (Collins et al., 2005). The two-class variance ratio is used to rank each feature by how well it separates the sample distributions of object and background. Top N features that have the greatest discrimination are selected to embed in a mean-shift tracking system. This approach,however only considers the RGB colour features. Actually, this approach is one feature-based tracking. Furthermore, it lacks an effective model update method to copy with the model drifts. Liang et al. extend the work of Collins et al. by introducing adaptive feature selection and scale adaptation (Dawei and Qingming, 2007). A new feature selection method based on Bayes error rate is proposed. But how to deal with the model drifts is not addressed in this paper. He et al. used a clustering method to segment the object tracking according to different colours, and generate a Gaussian model for each segment respectively to extract the colour feature (Wei and Xiaolin, 2007). Then an appropriate model was selected by judging the discrimination of the features. The Gaussian model however, not always fit each segment in practise. Recently, Wang and Yagi extended the standard mean-shift tracking algorithm to an adaptive tracker by selecting reliable features from RGB, HSV, normalized RG colour cues and shape-texture cues, according to their descriptive ability (Wang and Yaqi, 2008). But only two best discriminate features are used to represent the target. It dose not use fully all the features information it has computed and has a high time complexity.

A key issue addressed in this work is an online, adaptive multiple-feature fusion and template-update mechanism for target tracking. Based on the theory of biologically visual recognition system, the main visual information comes from the colour feature, edge feature and the texture feature (Thomas and Gabriel, 2007; Jhuang et al., 2007; Bar and Kassam, 2006). In this paper, RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are employed to implement the scheme. Target tracking is considered as a local discrimination problem with two classes: foreground and background. Many works have point the features that best discriminate between object and background are also best for tracking performance (Collins et al., 2005; Thomas and Gabriel, 2007; Jhuang et al., 2007). In this paper the tracked target is represented by a fused feature. According to the discriminate between object and background measured by two-class variance ratio, the multiple features are combined by linear weighting to realize kernel-based tracking. As model drafts, better performance could be achieved by using a novel up-dating strategy that takes into account the similarity between the initial and current appearance of the target. Each feature's similarity is computed. The high similarity features are given a big weight and the low similarity features are given a small weight. A good feature for tracking is a steady feature across the consecutive frames. The target update model is updated by re-weighting the multiple features based on the similarity between the initial and current appearance of the target. The proposed approach is shown as Figure 1.

The paper is organized as follows. Section 2 presents a brief introduction the feature extraction. Section 3 presents our proposed approach for target tracking. Computer simulation and results compared with related work are presented in Section 4. Concluding remarks are given in Section 5.

2 FEATURE EXTRACTION

It is important to decide what kinds of features are used before constructing the feature fusion mechanism. In Collins et al. (2005), the set of candidate features is composed of linear combination of RGB pixel. In Wang and Yaqi (2008), colour cue and shape-texture cues are employed to describe the model of the target. In this paper based on the theory of biologically visual recognition system (Thomas and Gabriel, 2007; Jhuang et al., 2007; Bar and Kassam, 2006), RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are employed to implement the scheme.

2.1 RGB Colour Feature

Colour information is an important visual feature. That is robust to the target rotary, non-rigid transformation and target shelter, widely used in the appearance model-based visual application. In this paper, colour distributions are represented by colour histograms, and RGB colour space as a very common colour space is used in this paper. The R, G and B channels are quantized into 256 bins, respectively. The colour histogram, calculated using Epanechnikov kernel, is applied (Comaniciu and Meer, 2002)(Comaniciu et al., 2003).

2.2 Prewitt Edge Feature

The edge information is the most fundamental characteristic of images. It is also included useful information for target tracking. There are many methods

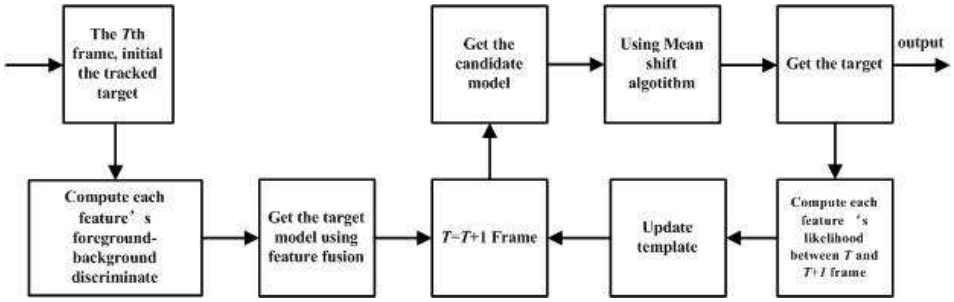


Figure 1: The Flow chart of the purposed approach.

for edge detection, but most of them can be grouped into two categories: search-based and zero-crossing based. The search-based methods detect edges by first computing a measure of edge strength, usually a first-order derivative expression, such as the gradient magnitude; and then searching for local directional maxima of the gradient magnitude using a computed estimate of the local orientation of the edge, usually the gradient direction (Yong and Croitoru, 2006; Sharifi et al., 2002). The zero-crossing based methods search for zero crossings in a second-order derivative expression computed from the image, in order to find edges, usually the zero-crossings of the Laplacian or the zero-crossings of a non-linear differential expression (Sharifi et al., 2002).

In this paper, Prewitt operator is employed to get the edge feature. For its low computational complexity and high performance. Prewitt operator has two convolution kernels as shown in Figure 2. Images of each point are used for the convolution kernel, the first kernel usually corresponding to the largest vertical edge, and the second corresponding to the largest horizontal edge. The maximum values of each point convoluted with the two kernels are accepted. Convolution is admitted as output value; results of operations are edge images. The Prewitt operator can be defined as

$$S_p = \sqrt{d_x^2 + d_y^2}, \tag{1}$$

$$\begin{aligned} d_x =& [f(x-1,y-1) + f(x,y-1) + f(x+1,y-1)] - \\ & [f(x-1,y+1) + f(x,y+1) + f(x+1,y+1)], \end{aligned} \tag{2}$$

$$\begin{aligned} d_y =& [f(x+1,y-1) + f(x+1,y) + f(x+1,y+1)] - \\ & [f(x-1,y-1) + f(x-1,y) + f(x-1,y+1)]. \end{aligned} \tag{3}$$

The histogram is used to represent the edge feature. Prewitt edge is also quantized into 256 bins Epanechnikov kernel like colour feature.

1	1	1
0	0	0
-1	-1	-1

1	0	-1
1	0	-1
1	0	-1

Figure 2: The two convolution kernel of Prewitt.

2.3 LBP Texture Feature

Local Binary Patterns (LBP) is basically a fine-scale descriptor that captures small texture details. It is also very resistant to lighting changes. LBP is a good choice for coding fine details of facial appearance and texture (T and T, 2007)(Aroussi and Mohamed, 2008)(Ojala et al., 1996). The Local Binary Patterns operator is introduced as a means of summarizing local gray-level structure by Ojiala in 1996 (Ojala et al., 1996). The operator takes a local neighbourhood around each pixel, thresholds the pixels of the neighbourhood at the value of the central pixel, and uses the resulting binary-valued image patch as a local image descriptor. It was originally defined for 3*3 neighbourhoods, given 8 bit codes based on the 8 pixels around the central one. Formally, the LBP operator takes the form

$$LBP(x_k, y_k) = \sum_{n=0}^{7} 2^n S(i_n - i_k), \tag{4}$$

$$S(u) = \begin{cases} 1, & u \geq 0, \\ 0, & u < 0, \end{cases} \tag{5}$$

where in this case n runs over the 8 neighbours of the central pixel k, i_k and i_n are the gray-level values at k and n, and $S(u)$ is 1 if $u \geq 0$ and 0 otherwise. The LBP encoding process is illustrated in Figure 3.

In methods that turn LBPs into histograms, the number of bins can be reduced significantly by assigning all non-uniform patterns to a single bin, often without losing too much information. In this paper, it is quantized into 256 bins with Epanechnikov kernel.

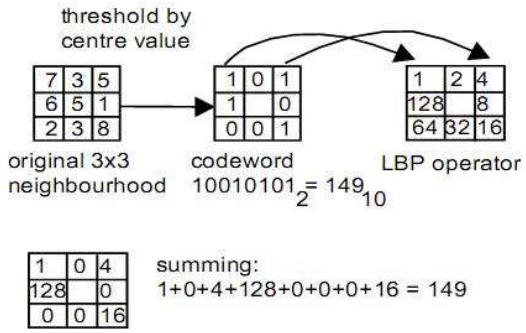


Figure 3: Illustration of the basic LBP operator.

3 ONLINE, ADAPTIVE FEATURES FUSION METHOD FOR TRACKING

There are two main components in this approach: the online adaptive features fusion based on discrimination criterion function, and the kernel-based tracking, which is used to track targets, based on the fused feature.

3.1 Features Fusion Method

In this paper, the target is represented by a rectangular set of pixels covering the target, while the background is represented by a larger surrounding ring of pixels. Given a feature f, let $H_{fg}(i)$ be a histogram of target and $H_{bg}(i)$ be a histogram for the background. The empirical discrete probability distribution $p(i)$ for the object and $q(i)$ for the background, can be calculated as $p(i) = H_{fg}(i)/n_{fg}$ and$q(i) = H_{bg}(i)/n_{bg}$, where n_{fg} is the pixel number of the target region and n_{bg} the pixel number of the background. The weight histograms represent the features only. It does not reflect the descriptive ability of the features directly. A log-likelihood ratio image is employed to solve this problem [14, 15]. The likelihood ratio nonlinear log likelihood ratio maps feature values associated with the target to positive values and those associated with the background to negative values. The likelihood ratio of a feature is given by

$$L(i) = \max(-1, \min(1, log(\frac{\max(p(i), \varepsilon)}{\max(q(i), \varepsilon)}))), \quad (6)$$

where ε is a very small number(set in 0.001 in this work), that prevents dividing by zero or taking the log of zero. Likelihood ratio images are the foundation for evaluating the discriminative ability of the features in the candidate features set. Figure 4 shows the likelihood ratio images of different features.

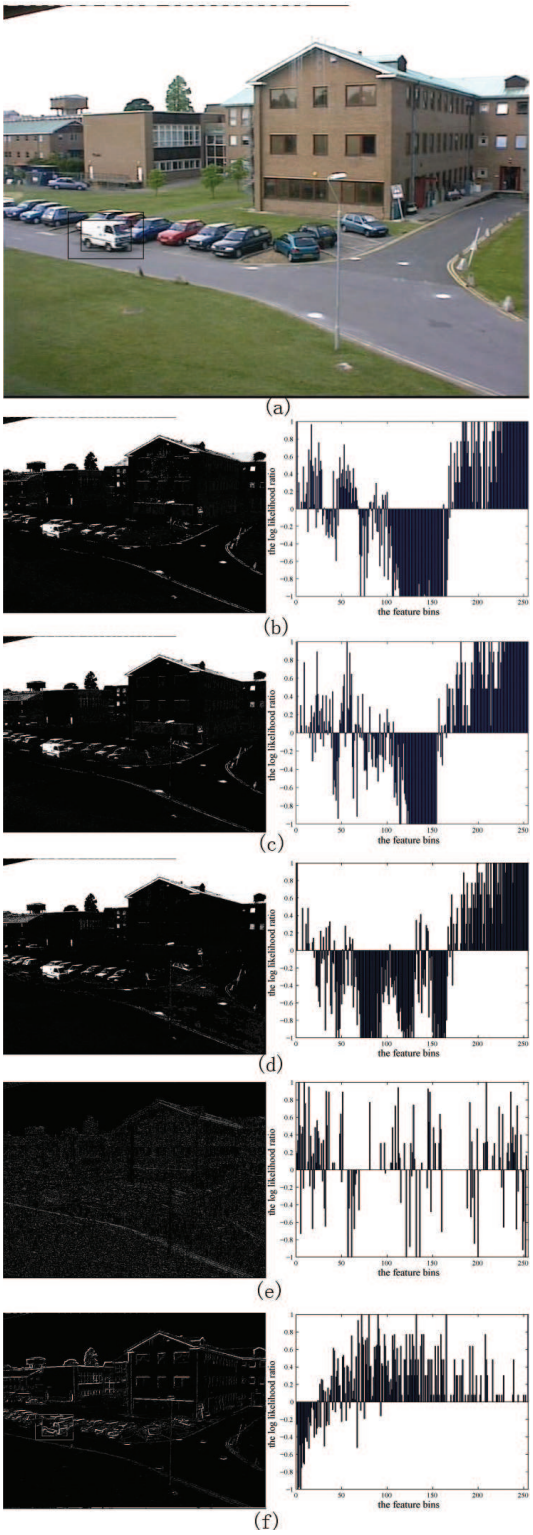


Figure 4: Different feature images after likelihood ratio process. (a) The input frame; (b) feature R and likelihood ratio; (c) feature G and likelihood ratio; (d) feature B and likelihood ratio; (e) LBP texture feature and likelihood ratio; (f) Prewitt edge feature and likelihood ratio.

In the practise, the whole weighted images weighted by log likelihood are not needed to be calculated for the computational complexity. The corresponding variance is employed to measure the separately between target and background classes. Using the method in Collins et al. (2005) and Wang and Yaqi (2008). based on the equality $var(x) = E[x^2] - (E[x])^2$, the variance of the log likelihood is computed as

$$\mathrm{var}(L:p) = E[L(i)^2] - (E[L(i)])^2. \quad (7)$$

The discriminative ability of each feature is calculated by the variance ratio. The hypothesis in this paper is that the features that best discriminate between target and background are also best for tracking the target. So, as the target's features describe model, a highest weight is given to the best discriminate feature, and the less discriminate feature has a smaller contribution. Based on the discrimination criterion function , the target features describe model can be calculated as

$$p_t(i) = \sum_{k=1}^{n} \lambda_k p_k(i), \quad (8)$$

$$\lambda_k = \frac{var_k(L:p)}{\sum_{k=1}^{n} var_k(L:p)}, \quad (9)$$

where $p_k(i)$ is the feature $K's$ probability distribution model, $\lambda_k$is the weight and $\sum_{k=1}^{n} \lambda_k = 1$. Figure 5 shows the fusion of the five features image.

Figure 5: The image after fusion.

3.2 Kernel-based Tracking

Mean shift is a nonparametric kernel density estimator, which, based on the colour kernel density estimation, has recently gained more attention due to its low computational complexity and robustness to appearance change, however, the basic mean shift tracking algorithm assumes that the target representation is discriminative enough against the background. This assumption is not always true, especially when tracking is carried out in a dynamic background (Comaniciu et al., 2003)(Li et al., 2008). An online, adaptive features fusion mechanism is embedded in the kernel-based mean shift algorithm for effective tracking. Due to the continuous nature of video, the distribution of target and background features in the current frame should remain similar to the previous frame and the fused feature model should still be valid. The initial position of the target is given by $y0$ which is determined in the previous frame. The target model is $P = p_{t,t=1\ldots m}, \sum_{t=1}^{m} p_t = 1$, and the candidate target model is $P(y_0) = p_t(y_0)_{t=1\ldots m}, \sum_{t=1}^{m} p_t = 1$, where p_t is the fused feature model. The Epanechnikov profile [8, 9] is employed in this paper. The target's shift vector form y_0 in the current frame is computed as

$$y_1 = \frac{\sum_{i=1}^{n_h} X_i \omega_i g(\|\frac{y_0 - X_i}{h}\|^2)}{\sum_{i=1}^{n_h} \omega_i g(\|\frac{y_0 - X_i}{h}\|^2)}, \quad (10)$$

where $g(x) = -k'(x)$,$k(x)$ is Epanechnikov profile, h is bandwidth and $\omega_i$can compute as

$$\omega_i = \sum_{t=1}^{m} \sqrt{\frac{P}{P(y_0)}} \delta[b(x_i) - t]. \quad (11)$$

The tracker assigns a new position of the target by using

$$y_1 = \frac{1}{2}(y_0 + y_1). \quad (12)$$

If $\|y_0 - y_1\|$, the iteration computation stops and y_1 is taken as the position of the target in the current fame. Otherwise let $y_0 = y_1$, then using Eq. (10) get the shift vector and do position assignment using Eq. (12). From Eq. (8) and Eq. (10), the pixels' weight is assigned by two parts. One is the kernel profile, which gives high weight to the pixel nearly to the centre. The other one is the discriminative ability of each feature. Higher weight is given to the higher discriminative ability feature.

3.3 Template Update Mechanism

It is necessary to update the target model, because the variation of the target appearance and its background. When the target appearance or the background changes, the fixed target model can not accurate describe the target, so it can not obtain the right position of target. But using an inaccurately tracking result to update the target model may lead to the wrong update of the target model. With the error accumulate, it finally results in track failure.

In order to alleviate the mode drifts, an adaptive model update mechanism based on the likelihood of

the features between successive frames is proposed in this paper. During the initialization stage, the target is obtained by a hand-draw rectangle and the target model is computed by the fusion method introduced in the previous subsection. The fused target model is used for tracking in the next frame and is also kept to use in subsequent model updates. Following the method in (Wang and Yaqi, 2008), the updated target model M can be computed as

$$M = (1-L_{ic})M_i + L_{ic}M_c, \quad (13)$$

$$L_{ic} = \sum_{u=1}^{m} \sqrt{M_i M_c}, \quad (14)$$

where L_{ic} is the likelihood between the initial model and current model measured by Bhattacharyya coefficient (Comaniciu and Meer, 2002)(Comaniciu et al., 2003); M_i is the initial model; M_c is the current target model computed as

$$M_c = (1-L_{pc})M_p + L_{pc}M_a, \quad (15)$$

where M_p is the previous target model, M_a is the current target fused-feature introduced in section 3, and L_{pc} is computed the likelihood between the M_a and M_p. The proposed updating method considers temporal coherence by weighing the initial target model, previous target model and current candidate. It can be more robust for the target appearance and the background change.

4 EXPERIMENTS

To illustrate the benefits of the proposed approach, experiments on various test video sequences using the proposed approach and other algorithms are conducted. The experiment was done using Pentium core 1.8G, Win XP, MATLAB 7.0. The Epanechnikov profile was used for histogram computations. The RGB colour feature, LBP texture feature and Prewitt edge feature were taken as feature space and it was quantized into 256 bins. The public dataset with ground truth is used to test the proposed method (Collins et al., 2005). The tracking results are compared with the basic mean-shift (Comaniciu and Meer, 2002), fore/background ratio, variance ratio, peak difference (Comaniciu et al., 2003), and multiple feature (Wang and Yaqi, 2008) trackers. The initialization of the tracking is given by a hand-draw rectangle. The same initializations are given to all the trackers. In this experiment, sequences EgTest01, EgTest02, EgTest03, EgTest04 and EgTest05 in the database are chosen. This challenging tracking sequences is made by various factors, such as different viewpoints, illumination changes, reflectance variations of the targets, similar objects nearby, and partial occlusions.

The tracking success rate is the most important criterion for target tracking, which is the number of successful tracked frames divided by the total number of frames. The bounding box that overlaps the ground truth can be considered as a successful track. To demonstrate the accuracy of tracking, the average overlap between bounding boxes (Avg overlap BB) and average overlap between bitmaps within the overlapping bounding box area (Avg overlap BM) are employed. Avg overlap BB is the percentage of the overlap between the tracking bounding box and the bounding box identified by ground truth files. Avg overlap BM is computed in the area of the intersection between the user bounding box and the ground truth bounding box. The comparison results are shown in Table 1.

From the comparison results that show the successful tracking ratio, the proposed tracker gives the best results in five of the test sequences. The basic mean-shift tracker dose not have a good performance in EgTest01, EgTest02 and EgTest05, because the basic mean-shift tracker dose not use the multi-features information and lacks adaptive strategy. Although an adaptive strategy is employed in Collins's approach, it dose not have good performance in EgTest01, EgTest02 and EgTest03. The peak difference algorithm has a better performance in the first sequences. The others sequences, however, do not demonstrate a good performance either. The multi-features methods that Integrate Colour and Shape-Texture features have a higher performance in all the sequences. Although the proposed approach has the best performance than the other trackers. But in EgTest03, EgTest04 and EgTest05 the successful tracking ratio is only 25.30%, 12.03% and 24.31%. The failed tracking examples are show as Figure 6.The main reason leading to tracking failure includes the similar feature distribution nearby as (a) in Fig. 6, the lower discrimination between foreground and background as (b) in Fig. 6 and long time occlusions as (c) in Fig. 6.

For accuracy of tracking, the proposed tracking algorithm is not the best in some of the sequences. There is not obvious correlation between the tracking accuracy criterion and the tracking successful ratio.The proposed approach does not have the highest accuracy, because in most frame-sequences,the background and the target are not always separated accurately.

Table 1: Tracking performance of different algorithms. (a) EgTest01; (b) EgTest02; (c) EgTest03; (d) EgTest04; (e) EgTest05.

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	17.58%	100%	29.12%	100%	100%	100%
Avg overlap BB	65.50%	62.87%	76.87%	61.76%	61.62%	74.28%
Avg overlap BM	66.26%	49.15%	61.30%	57.76%	68.38%	76.56%

(a)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	39.23%	39.23%	27.69%	30.77%	100%	100%
Avg overlap BB	91.09%	89.13%	85.19%	90.54%	93.32%	94.21%
Avg overlap BM	74.69%	66.98%	73.32%	65.91%	72.70%	73.53%

(b)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	20.62%	17.90%	12.06%	12.06%	20.23%	25.30%
Avg overlap BB	86.96%	87.01%	93.74%	92.27%	88.66%	90.11%
Avg overlap BM	66.65%	54.04%	70.79%	67.20%	69.37%	71.23%

(c)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	9.84%	8.74%	9.84%	3.83%	9.84%	12.03%
Avg overlap BB	66.78%	67.92%	66.03%	63.60%	69.52%	71.00%
Avg overlap BM	59.70%	52.75%	66.74%	66.42%	56.34%	68.43%

(d)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	13.64%	13.64%	13.64%	13.64%	21.22%	24.31%
Avg overlap BB	94.58%	88.75%	86.46%	86.98%	72.65%	75.41%
Avg overlap BM	84.02%	71.12%	85.12%	69.90%	64.45%	70.03%

(e)

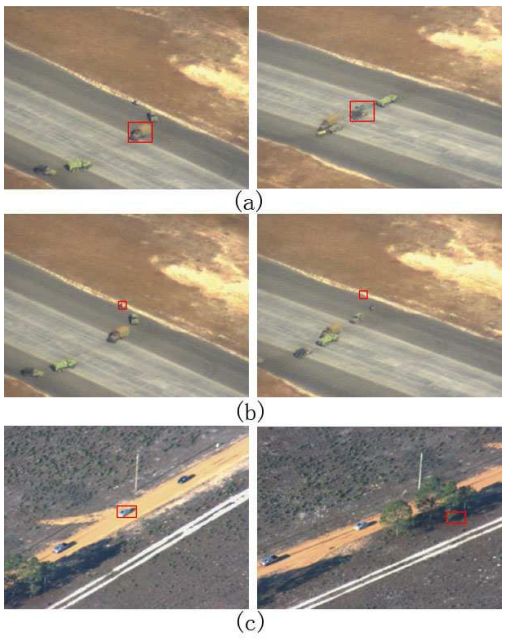


Figure 6: The failure tracking examples of the proposed approach. (a) and (b) the failure frame in EgTest03; (c) the failure result in EgTest04.

5 CONCLUSIONS

An online, adaptive multiple-feature fusion and template update mechanism for kernel-based target tracking is presented in this paper. Based on the theory of biological visual recognition system, RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are utilized to implement the proposed scheme. Experiment results show that the proposed approach is effective in target tracking. The comparison studies with other algorithms show that the proposed approach performs better in tracking of moving targets.

ACKNOWLEDGEMENTS

This work is partially supported by China Scholarship Council.

REFERENCES

Aroussi, E. and Mohamed (2008). Combining dct and lbp feature sets for efficient face recognition. In *3rd Inter-*

national Conference on Information and Communication Technologies: From Theory to Applications, 3rd International Conference on Information and Communication Technologies: From Theory to Applications, pages 1–6.

Bar, M. and Kassam, K. (2006). Top-down facilitation of visual recognition. In *Proceedings of the National Academy of Sciences*, volume 103 of *Proceedings of the National Academy of Sciences*, pages 449–454.

Barron, J. and Fleet, D. (1994). Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77.

Chang, W., Chen, C., and Jian, Y. (2008). Visual tracking in high-dimensional state space by appearance- guided particle filtering. *IEEE Transactions on Image Processing*, 17(7):1154–1167.

Collins, R., Lipton, A., Fujiyoshi, H., and Kanade, T. (2001). Algorithms for cooperative multi-sensor surveillance. In *Proceedings of the IEEE*, volume 89 of *Proceedings of the IEEE*, pages 1456–1477.

Collins, R., Liu, Y., and Leordeanu, M. (2005). Online selection of discriminative tracking features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1631–1643.

Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619.

Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577.

Dawei, L. and Qingming, H. (2007). Mean-shift blob tracking with adaptive feature selection and scale adaptation. In *IEEE Conference on Image Processing*, volume 3 of *IEEE Conference on Image Processing*, pages 369–372.

Jhuang, H., Serre, T., Wolf, L., and Poggio, T. (2007). A biologically inspired system for action recognition. In *IEEE 11th international conference on Computer Vision*, volume 14 of *IEEE 11th international conference on Computer Vision*, pages 1–8.

Li, Z., Tang, L., and Sanq, N. (2008). Improved mean shift algorithm for occlusion pedetrian tracking. *Electronics Letters*, 44:622–623.

Maggio, E., Smerladi, F., and Cavallaro, A. (2007). Adaptive multifeature tracking in a particle filtering framework. *IEEE transactions on Circuits and Systems for Video Technology*, 12(10):1348–1359.

Mattews, I., Iashikawa, T., and Baker, S. (2004). The template update problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):810–815.

Ojala, T., Pietikainen, M., and Harwood, D. (1996). A comparative study of texture measures with classifcation based on feature distributions. *Pattern Recognition*, 29(1):51–59.

Pan, J., Ho, B., and Zhang, J. Q. (2008). Robust and accurate object tracking under various types of occlusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(2):223–236.

Shalom, Y. and Fortmann, T. (1988). *Tracking and Data Association*. Academic Press, Boston.

Sharifi, M., Fathy, M., and Mahmoudi, M. (2002). A classified and comparative study of edge detection algorithms. In *International Conference on Information Technology: Coding and Computing*, International Conference on Information Technology: Coding and Computing, pages 117–120.

Stauffer, C. and Grimson, W. (1995). Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):814–820.

T, X. and T, B. (2007). *Fusing Gabor and LBP Feature Set for Kernel-Based Face Recognition*, volume 4778 of *Lecture Notes in Computer Science*. Springer Berlin, Heidelberg.

Tal, N. and Bruckstein, A. (2008). Over-parameterized variational optical flow. *International Journal of Computer Vision*, 76(2):205–216.

Thomas, S. and Gabriel, K. (2007). A quantitative theory of immediate visiual recognition. *Program on Brain Research*, 167:33–56.

Wang, J. and Yaqi, Y. (2008). Integrating color and shape-texture features for adaptive real-time object tracking. *IEEE Transactions on Image Processing*, 17(2):235–240.

Wei, H. and Xiaolin, Z. (2007). Online feature extraction and selection for object tracking. In *IEEE Conference on Mechatronics and Automation*, IEEE Conference on Mechatronics and Automation, pages 3497–3502.

Yong, Y. and Croitoru, M. (2006). Nonlinear multiscale wavelet diffusion for speckle suppression and edge enhancement in ultrasound images. *IEEE transactions on Medical Imaging*, 25(3):297–311.

Zhaozhen, Y., Poriki, F., and Collins, R. (2008). Likelihood map fusion for visual object tracking. In *IEEE workshop on Applications of Computer Vision*, IEEE workshop on Applications of Computer Vision, pages 1–7.

FROM BENDING TO LINEAR MOVEMENT
A Linear Actuation Mechanism based on Conducting Polymer Actuators

Elise T. Burriss, Gursel Alici
University of Wollongong, School of Mechanical, Materials and Mechatronic Engineering, Australia
etb984@uow.edu.au, gursel@uow.edu.au

Geoffrey M. Spinks, Scott McGovern
University of Wollongong, School of Mechanical, Materials and Mechatronic Engineering, Australia
gspinks@uow.edu.au, scottmg@uow.edu.au

Keywords: Electroactive polymer actuators, Linear actuation, Kinematic design.

Abstract: The objects of this study are (i) to establish a linear actuation system based on the bending movement of conducting polymer actuators, which operate in air, and (ii) to develop a model to predict the linear displacement and force output of the actuation system, and employ the model to characterise the optimum values of the system design parameters. The linear actuation system is based on a five bar parallel mechanism with equal link lengths and a zero-length ground link, which is articulated through two bending type polymer actuators. Kinematic and force analyses of the mechanism including numerical results are presented, and its payload handling ability was experimentally evaluated. Experimental results presented demonstrate that the linear actuation system can generate linear movement accurately and a rectilinear force as a result of two bending type polymer actuators. This mechanism is offered as a lightweight, low power consuming motion and force transmission system.

1 INTRODUCTION

Conducting polymers are favourable as actuators due to their low actuation voltage, high force output relative to their weight, good strain properties, light weight, simple structure and silent motion. The common materials used for actuators are Ionic Polymer Metal Composite (IPMC), and electroactive polymers (EAPs) such as Polypyrrole (PPy) and Polyaniline (PANI) (Thompson, 2007). Disadvantages of polymer actuators include low speed of response, cyclic fatigue and non-linearity, which increases difficulty in modelling and controlling the actuator (Alici, Metz & Spinks 2005, Eamex nd).

Bending actuators have been successfully used in applications where a nonlinear motion is required, such as a fin in an artificial fish (Alici et. al, 2007) and robotic fingers (Alici & Huynh, 2007). However, the bending motion may limit the control of actuators along a linear path. Conducting polymers have been used as actuators in applications such as medical devices, toys, digital camera accessories and artificial muscles for robots as developed by Eamex 2008. Various PPy linear actuator designs have been investigated, including the PPy linear design by Otero, Cortes & Vazquez Arenas (2007), in which two PPy films were connected in parallel and immersed in a $LiClO_4$ aqueous solution, were able to produce a displacement of 60% of the actuator length in the longitudinal direction of the actuator. The use of linear actuators in air may provide an alternative when encapsulation of the electrolyte is not feasible. Yamakita et. al (2004) have developed an IPMC linear actuator for use in a biped walking robot. PPy actuators are used in applications where the oxidised or reduced position is maintained under a constant voltage, for which IPMC is not suitable as it will return to its original position. To overcome this limitation of IPMC, Thompson (2007) adopted the design by Yamakita et al. using polypyrrole film as the active layers of a multi-layer actuator structure. Two methods of connecting the PPy film together to produce the linear design were employed by Thompson: (i) masking part of the film during PPy growth to create a flexible, electrically conductive hinge; (ii) electrically connecting the film with Cu

tape, held in place with NdFeB magnets. This study implements the design investigated by Thompson using the second method of connecting the film together, as the method of masking allows the actuator to produce a clover-like shape under actuation, which impedes control and stability of the actuator. However, limitations arising through the use of Cu tape may include oxidation of the Cu, converting the tape to an insulator, thus limiting the actuation of the PPy components connected by copper tape. For optimisation of the PPy film in this study, the geometry of PPy bending actuators is limited to 50μm thickness and 4mm width as the charge distribution has been found to be no longer uniform at thicknesses greater than 50-60μm and curling is prominent at widths greater than 4mm, producing an increase in elastic stiffness of the actuator (Alici, Metz & Spinks 2005, Metz, Alici & Spinks 2006). These limitations are expected to be also evident in PPy linear actuators.

A significant amount of work to determine the performance of bending PPy actuators has been previously conducted (Alici & Huynh 2007, Metz, Alici & Spinks 2006, John, Alici & Cook 2008). However, the complexity and greater degree of variability in a linear actuator means that the performance of a bending actuator may not directly translate to linear actuators. The modelling of linear actuators is necessary to accurately predict the force, displacement and work outputs for control of the actuator in practical applications. Modelling and optimisation work on PPy bending actuators by Alici, Metz & Spinks (2005), in which a mathematical model to calculate the expected bending behaviour of PPy bending actuators was developed, and suggests that as the length of the actuator decreases, force output increases. This study has investigated the effect of the length of PPy components in linear actuators to determine if length has a similar effect on outputs in linear actuators. Combinations of 2mm, 3mm, 4mm wide and 10mm, 15mm and 20mm long linear actuators, where the length refers to the length of each component, were experimentally evaluated for force and displacement outputs. It was found that an increase in the length of the PPy links in the linear actuators allowed a greater range of motion, whilst shorter and wider actuators lifted greater loads. The linear actuators were also compared to bending actuators of corresponding dimensions and it was found that the bending actuators had a greater range of motion and the linear actuators lifted heavier loads. As the proposed linear actuation mechanism converts the rotational work into the linear work, ideally they are equal to each other, it is expected that while the displacement is decreasing, the force is increasing, and vice versa.

2 FUNCTIONING PRINCIPLE OF LINEAR ACTUATION SYSTEM

As the polymer actuators considered in this study are cantilevered from one end, they generate a rotary (bending) type motion. This is analogous to a single jointed crank motion. A single degree of freedom mechanism such as a four-bar mechanism does not allow control the output motion. With this in mind, a five-bar mechanism which needs two inputs (two bending type polymer actuators) is one of the multi crank mechanisms having practical importance, especially for following any arbitrary trajectory precisely (Alici, 2000). This mechanism serves as a bending to-linear motion converter; converting bending angle into a linear movement, and bending moments into linear forces.

2.1 Kinematic Analysis

The topology of the mechanism is depicted in Figure 1, where it is assumed that the mechanism links are rigid for the sake of generating a pseudo-rigid-body-model for the size and operation optimization of this linear motion mechanism. The joints connected to the frame are the active ones. The others are passive. For the joint inputs θ_1 and θ_2 and the specified link lengths L_0, L_1, L_2, L_3, L_4 the analytical expressions for the coordinates of the output point **P** are obtained using the following algorithm (Alici, 2000):

$$\begin{aligned} \vec{L}_1 &= L_1 \cos\theta_1 \, \vec{i} + L_1 \sin\theta_1 \, \vec{j}, \\ \vec{L}_5 &= \left(L_0 + L_4 \cos\theta_2\right)\vec{i} + L_4 \sin\theta_2 \, \vec{j}, \end{aligned} \tag{1}$$

$$\left|AB\right| = \vec{R} = \left|O_1B\right| - \vec{L}_1 = C\,\vec{i} + D\,\vec{j}, \tag{2}$$

$$\begin{aligned} &C = \left|O_1B\right|_x - L_{1x}, \quad D = \left|O_1B\right|_y - L_{1y}, \\ &R = \sqrt{C^2 + D^2}, \end{aligned} \tag{3}$$

$$Q = \frac{L_2^2 + R^2 - L_3^2}{2R^2}, \tag{4}$$

The coordinates (x_P, y_P) of the output point P are;

$$x_P = x_A + Q\,C - D\sqrt{\frac{L_2^2}{R^2} - Q^2}, \qquad y_P = y_A + Q\,D + C\sqrt{\frac{L_2^2}{R^2} - Q^2}, \tag{5}$$

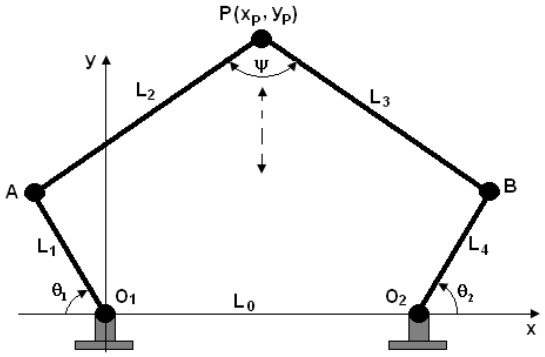


Figure 1: Proposed linear actuation module based on a parallel five-bar mechanism.

Depending on the link lengths, the expression in the square-root can become negative, which suggest that some kinematic design constraints are not satisfied. For a linear motion output, the output point can be constrained to move in a rectilinear fashion. This requires that $\theta_1 = \theta_2$ AND $L_1 = L_4$ AND $L_2 = L_3$. Such a mechanism will convert all of its rotary work into the linear work - an efficient motion converter. The analytical expressions for the output point of such a linear motion module are obtained as

$$\begin{aligned} x_P &= -L_1\cos\theta_1 + \frac{1}{2}\left[L_0 + L_1(\cos\theta_2 + \cos\theta_1)\right] \\ &\quad - L_1(\sin\theta_2 - \sin\theta_1)\sqrt{\frac{L_2^2}{R^2} - \frac{1}{4}} \\ y_P &= L_1\sin\theta_1 + \frac{L_1}{2}(\sin\theta_2 - \sin\theta_1) \\ &\quad + \left[L_0 + L_1(\cos\theta_2 + \cos\theta_1)\right]\sqrt{\frac{L_2^2}{R^2} - \frac{1}{4}} \end{aligned} \tag{6}$$

where

$$\begin{aligned} R^2 &= L_0^2 + 2L_0L_1(\cos\theta_2 + \cos\theta_1) \\ &\quad + 2L_1^2\left[1 + \cos(\theta_1 + \theta_2)\right] \end{aligned} \tag{7}$$

Depending on the link length of the mechanism, Eq.6 simplifies to

i) For $L_1 = L_2$ and $\theta_1 = \theta_2 \Rightarrow x_P = \frac{L_0}{2}$ and

$$y_P = L_1\sin\theta_1 + \frac{1}{2}\sqrt{4L_1^2 - L_0^2 - 4L_0L_1\cos\theta_1 - 4L_1^2\cos^2\theta_1}$$

ii) For $L_0 = 0$ and $\theta_1 = \theta_2 \Rightarrow x_P = 0$ and $y_P = L_1\sin\theta_1 + \sqrt{L_2^2 - L_1^2\cos^2\theta_1}$

iii) For $L_0 = 0$, $L_1 = L_2$ and $\theta_1 = \theta_2 \Rightarrow x_P = 0$ and $y_P = 2L_1\sin\theta_1$

When choosing the link sizes, it is important for effective force transmission to minimize the variation of the transmission angle ψ from 90^0; the acceptable range is $90^0 \mp 40^0$ (Alici, 2004). It is mathematically expressed as

$$\cos\psi = 1 - \frac{1}{2}\left(\frac{R}{L_2}\right)^2 \tag{8}$$

For $50^0 \le \psi \le 130^0$, $0.845 \le \left(\frac{R}{L_2}\right) \le 1.8126$. With reference to Eq.8, for a given L_0 and L_1, the range of L_2 satisfying the effective transmission angles can be calculated. It must be noted that L_0 should be greater than L_1 to prevent any physical interference among the mechanism links. A practical ratios of $1.5 \le \left(\frac{L_0}{L_1}\right) \le 3.0$ and $1.0 \le \left(\frac{L_2}{L_1}\right) \le 2.0$ should result in proportionate link lengths and transmission angles. For the data shown in Table 1, the transmission angle and the vertical movement of the linear actuation module are calculated and presented in Figure 2.

Table 1: Numerical values of the parameters for the exemplary results shown in Figure 2.

Parameters
$L_0 = 40\,\text{mm}$
$L_1 = L_3 = 20\,\text{mm}$
$L_2 = L_4 = 22\,\text{mm}$
$\theta_1 = \theta_2 = 90^0$ to 115^0

The corresponding generalized relationship between rate of change $\dot{\Theta} = \left[\dot{\theta}_1 \quad \dot{\theta}_2\right]^T$ of the input bending angles and the output velocity vector

$\dot{\mathbf{X}} = \begin{bmatrix} \dot{x}_P & \dot{y}_P \end{bmatrix}^T$ is given by (Alici & Shirinzadeh, 2005)

$$\dot{X} = \begin{bmatrix} \dfrac{\partial x_P}{\partial(\theta_1, \theta_2)} & \dfrac{\partial y_P}{\partial(\theta_1, \theta_2)} \end{bmatrix}^T \dot{\Theta} = J\dot{\Theta} \qquad (9)$$

where J is the mechanism Jacobian matrix.

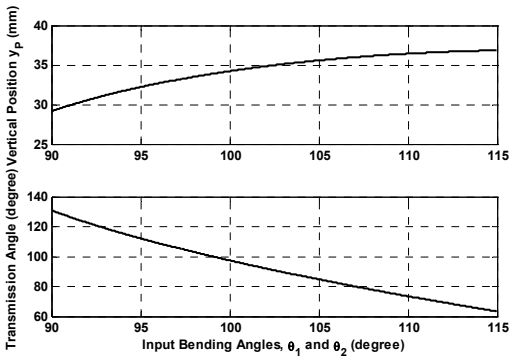


Figure 2: Variation of the vertical position of the output point P and the transmission angle with the input bending angles. The net vertical distance is 7.6858 mm.

2.2 Force Analysis

Assuming that two active links O_1A and O_2B made of electroactive polymers generate bending moments M_1 and M_2, which act on the actuation module in the opposite directions. Assume that the output point P can apply a planar force vector of $F_p = F_x\vec{i} + F_y\vec{j}$ to the environment to realise a functional task. Please recall that the mechanism converts the work in the bending coordinates Θ into the work in linear coordinates X. These two works are ideally equal to each other.

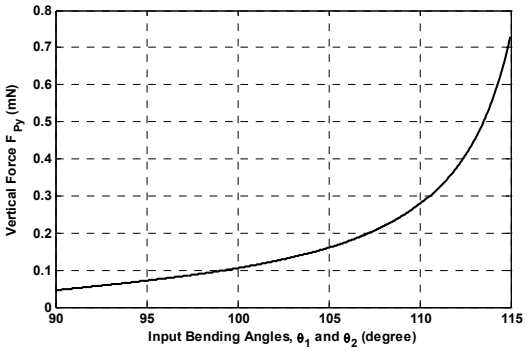


Figure 3: Variation of the vertical force component of the output point P with the input bending angles. The net vertical force is 0.6837 mN. The horizontal component of the force is zero.

Using the duality between the generalized relationships for motion and force transfer between the actuation and output spaces, the following force relationship is obtained (Alici & Shirinzadeh, 2005);

$$M = J^T F_P \;\Rightarrow\; F_P = J^{-T} M \qquad (10)$$

where $F_P = \begin{bmatrix} F_x & F_y \end{bmatrix}^T$ and $M = \begin{bmatrix} M_1 & M_2 \end{bmatrix}^T$ and $(.)^T$ denotes transposition. For the data in Table 1 and $M = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$ Nmm, the force output at point P is calculated using Eq.10 and is presented in Figure 3. Because the mechanism link lengths are chosen such that it can generate a vertical movement, the mechanism will not create any horizontal force component. These kinematic and force analyses suggest that it is possible to optimize the topology of the mechanism for an efficient motion and force transmission.

3 CONDUCTING POLYMER ACTUATORS: SYNTHESIS AND ACTUATION PRINCIPLE

The PPy film was manufactured by combining a solution of 0.1M pyrrole monomer with 0.1M lithium triflouromethanesulfonimide (Li^+TFSI^-) in propylene carbonate (PC) with 1 % H_2O, degassed with nitrogen gas and stirred for 15 minutes. This solution was used to grow the PPy onto a gold sputter coated, 0.45μm Millipore Immobilon-P porous poly(vinylidene fluoride) (PVDF) film by electrodeposition over a period of 12 hours at -33°C with a current density of 0.1mA.m^{-2}. After polymerization, the film was washed in acetone and then resoaked in a 0.1M LiTFSI electrolyte and then stored away from oxygen. Actuators were then cut to desired size using a scalpel and were limited to size by the area of the 40mm x 55mm film. The resulting film consists of two PPy outer layers and two gold layers separated by a PVDF core is shown in Figure 4.

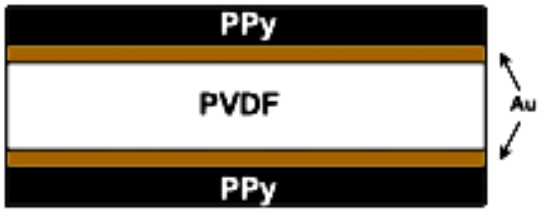


Figure 4: General structure of trilayer polymer actuator.

After doping with electrolyte, both the PPy layers of a bending actuator become partially oxidised (Figure 5a) and upon application of a positive voltage, one of the polymer layers is highly oxidized and the other is reduced. In an attempt to neutralise the charge imbalance, ions transfer from the electrolyte into the polymer layers. In LiTFSI doped actuators, $TFSI^-$ anions move into interstitial spaces in the polymer backbone of the oxidized layer, causing it to expand. Simultaneously, the opposing polymer film is reduced and contracts by removing $TFSI^-$ ions, overall creating a bending motion (Figure 5b.) By applying a square wave voltage, the actuator is allowed to return it to its original position and continue to bend through the neutral position so that the actuator may bend alternately in both directions.

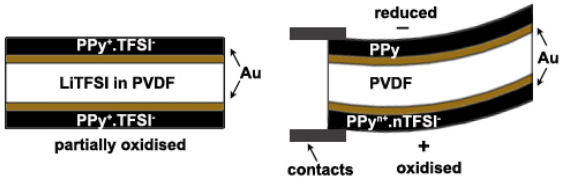


Figure 5: Schematic of PPy actuator doped in LiTFSI (a) partially oxidised (b) under an applied voltage.

To produce the linear actuators, two pieces of PPy film cut to equal dimensions were connected together in parallel at their tips with copper tape and two Ø 3mm × 0.5mm neodymium iron boron (NdFeB) magnets one either side of each tip (Figures 6-7.) Once placed between the electrodes, the PPy configuration effectively produces four PPy components, with the length referred to throughout this paper as the length depicted in Figure 7b.

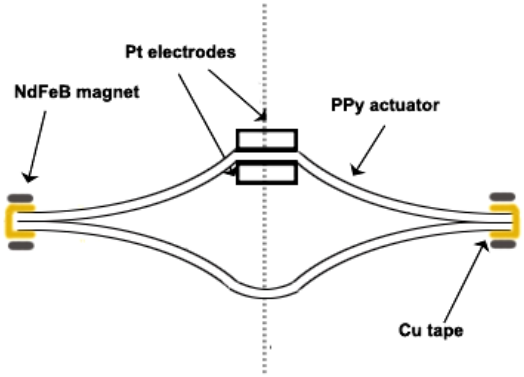


Figure 6: Configuration of the linear PPy actuator with Cu tape and NdFeB magnets.

Under an applied voltage, it was observed that the PPy linear actuator produced the shape as shown in Figure 8. In this case, the outer PPy layer of the upper components is oxidised and the inside layer is reduced, and under no constraints, almost uniform curvature may be experienced. The lower components may be activated; however the possible oxidation of the Cu tape suggests that the lower components remain either inactivated. This linear design restricts the bending motion of the actuators due to the presence of the magnets used to clamp the copper tape to the actuators (Figure 8). For the shorter lengths, a greater percentage of the actuator is affected by the clamping induced by the magnets and as a result, the actuator has a more elongated appearance (Figure 9). Due to this elongation, it doesn't move as far in the vertical direction as it potentially could. To overcome this limitation, another method of electrically connecting the upper and lower actuator components that does not restrict the motion is necessary. For this to be possible, the electrical connection must be flexible and able to bond to the actuator without the assistance of magnets. Although the magnets restrict motion, they do provide some stability which is important in attempting to control the actuator.

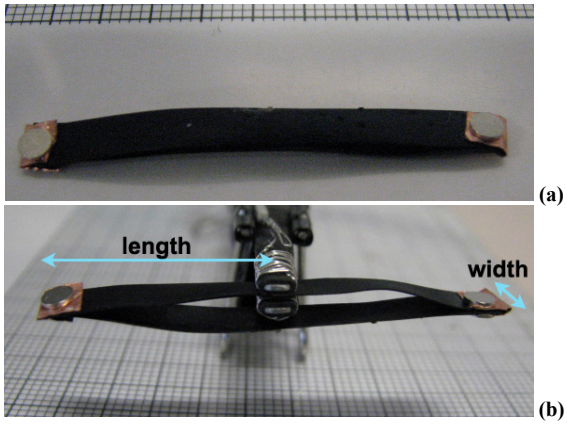


Figure 7: Linear PPy actuator in neutral position.

4 PERFORMANCE CHARACTERISATION RESULTS

Linear actuators shown in Figures 6-7 with link lengths of 10mm, 15mm, 20mm and widths of 2mm, 3mm, 4mm were manufactured and tested under a series of applied loads. Loads were added to the bottom tip of the actuators and the overall displacement of the base tip was measured visually using a grid paper, as illustrated in Figure 10a. The initial load used was the non-magnetic wire hook and a 0.266g NdFeB magnet glued to its base (Figure 10b), with incremental loads applied by

adding magnets of either 0.266g or 0.54g to the base of the wire hook. Voltage was maintained at +/-1V to avoid over-oxidation and reduce the variation in results that may arise due to early onset of fatigue. The frequency of the square voltage inputs was held constant for all experiments at 0.5Hz to allow sufficient response time. Typical voltage, current and charge data recorded by the datalogging system used is presented in Figure 11.

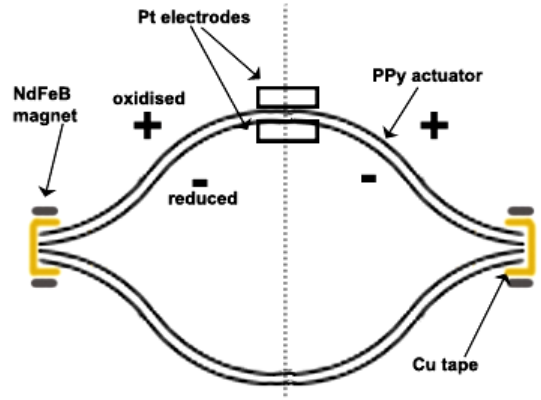


Figure 8: Shape of actuator under actuation.

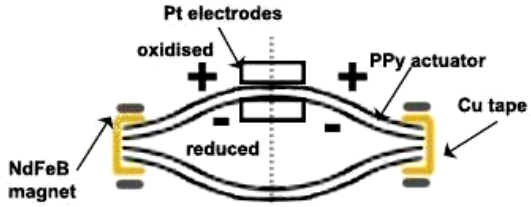
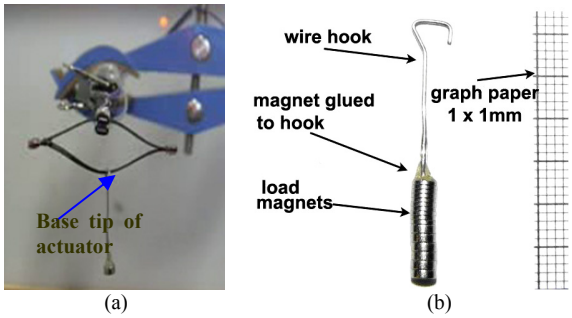


Figure 9: Shape of shorter actuators under actuation.



(a) (b)

Figure 10: (a) PPy linear actuator with a load. (b) Wire hook; used to separate magnet loads from magnets on the actuator.

Figures 12(a)-(i) provide an overview of the base tip position in extended and contracted positions under given applied loads, with the electrode contacts as the reference point. In all actuators, as the load increases the position of the base tip below the reference point also increases. Generally, as the width decreases the position below the electrode increases and is more evident in the 15mm long actuators. As width increases, the actuator becomes stiffer which allows a smaller deflection below the reference point under a load than an actuator of a small width.

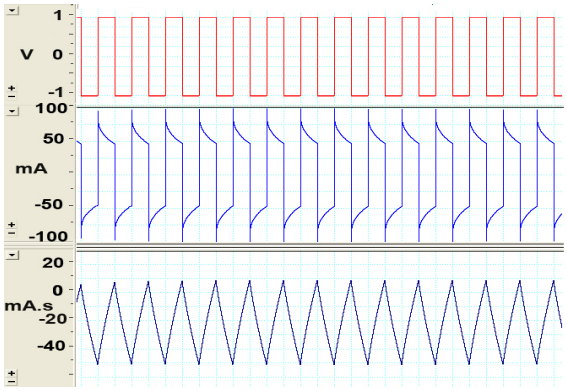


Figure 11: Example voltage, current and charge input data for 20mm x 2mm linear PPy actuator at 0.5Hz.

Displacement increases and then decreases for increasing load as shown in Figure 12(d)-(i) for the 15mm and 10mm long actuators. This behaviour is because under no load, the actuators are restricted in their movement by the magnets clamping the Cu tape in place. The clamping effect has a greater impact on shorter actuators because, as a percentage, a greater proportion of the actuator is restricted than longer actuators. Under applied loads, the shorter actuators are able to move because the load forces the actuator base tip to be displaced to lower than the original position, and under an applied voltage the actuator attempts to return to the original position and produces an upward motion. For this reason, the shorter actuators may only be suitable for applications requiring a pulling load, and if shorter actuators are required for pushing, then another method of connecting the actuators that provides less restriction on motion is required.

A comparison of load and displacement output between the linear and cantilever actuators of corresponding length has revealed that the cantilever actuators provide a greater range of motion in the vertical direction, while the linear actuator lifted heavier loads. The linear actuators are effectively 4 times the volume of their corresponding bending actuator.

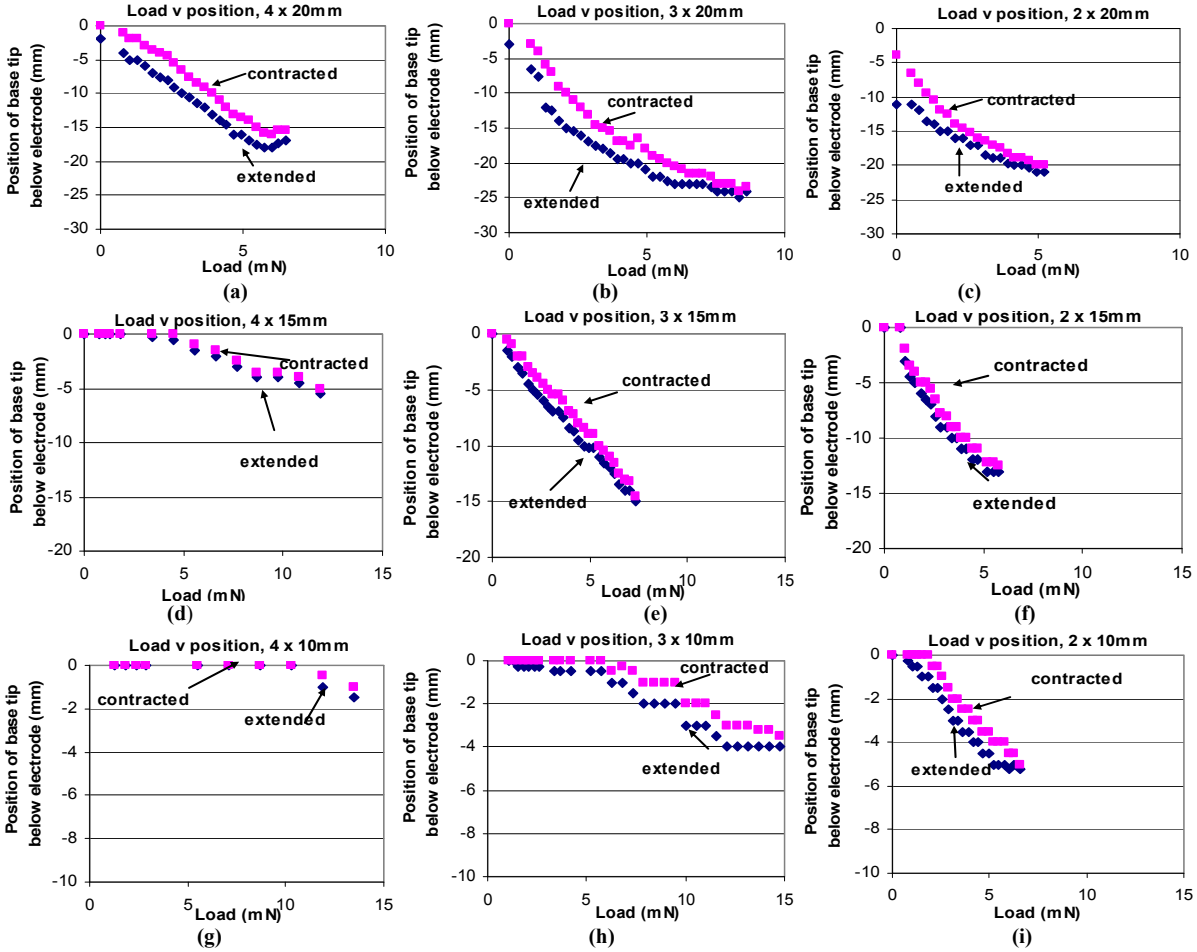


Figure 12: Position of base tip of linear actuators under applied load.

However, they produced much less displacement due to the restriction in motion provided by the magnet connections. The cantilever actuators have the ability to move above and below the electrodes whereas the linear actuators are limited to motion below the electrodes only. Although much less displacement is achieved, the benefit of a linear actuator is that motion in one direction is more controlled than for a bending actuator. For displacement output, the linear actuators with 20mm PPy links, produced the greatest displacement than actuators with shorter PPy links over a range of loads. Figure 13 depicts the displacement in the vertical direction for linear actuators of varied width with 20mm length. As width of the PPy links increases, a greater displacement output for a given load may be expected. However, as width increases, the PPy film has a tendency to curl across the width, particularly in actuators for 4mm wide or greater (Alici, 2007). Curling increases stiffness and thus decreases the potential displacement output, hence the 4mm wide actuator produced a slightly smaller displacement than actuator with 3mm wide PPy links.

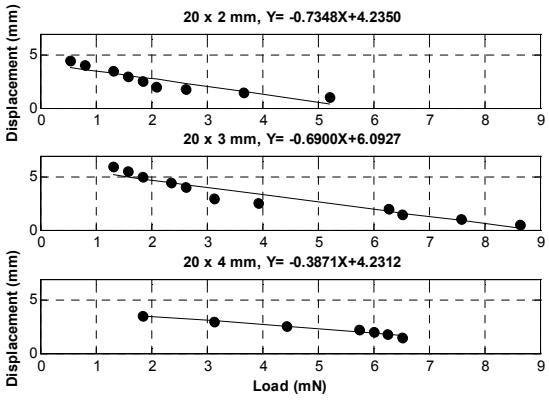


Figure 13: Variation of the tip displacement of the linear actuation module with the payload.

5 CONCLUSIONS

A linear actuator system based on the bending motion of conducting polymer actuators operating in air is presented, including an analytical model to estimate the linear movement and the force output of the mechanism. The mechanism is basically a motion and force transmission system, converting the bending work provided by the electroactive polymer actuators into Cartesian work. The experimental results presented demonstrate that the conducting polymer actuators generate enough displacement and force to handle a range of practical payloads. Another outcome of this study is that when the bending type- low power consuming polymer actuators are tailored properly, they can be used to generate a rectilinear motion with enough force output.

Future work involves deriving a more accurate analytical model taking into account the deflections of the mechanism links and verifying the model experimentally. Improvements may also be made to the hinge connections of the linear actuator, by replacing the copper connections with an inert, conductive material such as gold or platinum.

ACKNOWLEDGEMENTS

The authors thank Dr Stephen W. John for his help in synthesizing the bulk actuator sheet and construction of the actuation module.

REFERENCES

Alici, G 2000, 'Determination of Singularity Contours for Five-bar Planar Parallel Manipulators', *Robotica*, vol.18, no.6, pp.569-575.

Alici, G & Huynh, NN 2007, 'Performance Quantification of Conducting Polymer Actuators for Real Applications: A Microgripping System', *IEEE/ASME Transactions. on Mechatronics*, vol.12, no.1, pp.73 - 84.

Alici, G, Metz, P & Spinks, G 2005, 'A mathematical model to describe bending mechanics of Polypyrrole (PPy) Actuators', *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2005, 24-28 July, Monterey, California.

Alici, G & Shirinzadeh, B 2004 "Optimum synthesis of parallel manipulators based on kinematic isotropy and force balancing", Robotica, vol.22, no.1, pp. 97 – 108.

Alici, G & Shirinzadeh 2005, 'Enhanced Stiffness Modelling, Identification and Characterisation for Robot Manipulators', *IEEE Transactions on Robotics*, vol.21, no.4, pp. 554 – 564.

Alici, G, Spinks, GM, Huynh, NN, Sarmadi, L, & Minato, R 2007, 'Establishment of a Biomimetic Device Based on Tri-layer Polymer Actuators – Propulsion Fins', *Journal of Bioinspiration & Biomimetics*, vol.2, no.2, pp. S18-S30.

EAMEX Corporation n.d., accessed 15/06/2008, http://www.eamex.co.jp/index_e.html.

John, SW, Alici, G & Cook, CD 2008, 'Validation of a Resonant Frequency Model for Polypyrrole Trilayer Actuators', *IEEE/ASME Transactions on Mechatronics*, vol.13, no.4, pp.401 - 409.

Metz, P, Alici, G & Spinks, G 2006, 'A finite element model for bending behaviour of conducting polymer electromechanical actuators', *Sensors and Actuators A*, vol.130-131, pp1-11.

Otero, TF, Cortes, MT & Vazquez Arenas, G 2007, 'Linear movements from two bending triple-layers', *Electrochimica Acta*, vol. 53, pp1252-1258.

Thompson, F.W. 2007. 'Hybrid linear actuator based on conducting polymers', Bachelor of Engineering (Mechatronics) Honours Thesis, Faculty of Engineering, University of Wollongong.

Yamakita, M, Kamamichi, N, Yashuaki, K, Asaka, K & Luo, Z 2004 'Development of an artificial muscle linear actuator using ionic polymer metal composites', *Advanced Robotics,* vol.18 , pp383-399.

THE MULTISTART DROP-ADD-SWAP HEURISTIC FOR THE UNCAPACITATED FACILITY LOCATION PROBLEM

Lin-Yu Tseng[a,b] and Chih-Sheng Wu[b]
[a] *Institute of Networking and Multimedia, National Chung Hsing University, 250 Kuo Kuang Road, Taichung, Taiwan*
[b] *Department of Computer Science and Engineering, National Chung Hsing University*
250 Kuo Kuang Road, Taichung, Taiwan
lytseng@cs.nchu.edu.tw, tks368@ms22.hinet.net

Keywords: Facility location, Multistart MDAS heuristic, Heuristics, Optimization, Local search.

Abstract: The uncapacitated facility location problem aims to select a subset of facilities to open, so that the demands of a given set of customers are satisfied at the minimum cost. In this study, we present a novel multistart Drop-Add-Swap heuristic for this problem. The proposed heuristic is multiple applications of the Drop-Add-Swap heuristic with randomly generated initial solutions. And the proposed Drop-Add-Swap heuristic begins its search with an initial solution, then iteratively applies the Drop operation, the Add operation or the Swap operation to the solution to search for a better one. Cost updating rather than recomputing is utilized, so the proposed heuristic is time efficient. With extensive experiments on most benchmarks in the literature, the proposed heuristic has been shown competitive to the state-of-the-art heuristics and metaheuristics.

1 INTRODUCTION

The success of some businesses heavily depends on how they locate their facilities. Therefore, location problems have been widely studied because of their importance, both in theory and in practice. Location problems can be classified into four categories: p-center problems, p-median problems, uncapacitated facility location problems, and capacitated facility location problems. In this study, we consider the uncapacitated facility location problem (UFLP). A UFLP can be described as follows.

Suppose we have a set of m customers $U=\{1, 2, \ldots, m\}$ and a set of n candidate facility sites $F=\{1, 2, \ldots, n\}$. There is no limit to the number of customers a facility can serve. c_{ij} is used to represent the cost of serving customer i from facility j and f_j is used to represent the cost of opening facility j. Furthermore, c_{ij} for $i=1, 2, \ldots, m$ and $j=1, 2, \ldots, n$ and f_j for $i=1, 2, \ldots, n$ are assumed to be greater than zero. Then a UFLP is defined as (Cornuéjols et.al., 1990):

$$\text{Minimize} \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{j=1}^{n} f_j p_j$$

Subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \text{for} \quad i = 1,\ldots,m \tag{1}$$

$$x_{ij} \le p_j \quad \text{for} \quad i = 1,\ldots,m \quad \text{and} \quad j = 1,\ldots,n \tag{2}$$

$$x_{ij} \in \{0,1\} \quad \text{for} \quad i = 1,\ldots,m \quad \text{and} \quad j = 1,\ldots,n \tag{3}$$

$$p_j \in \{0,1\} \quad \text{for} \quad j = 1,\ldots,n. \tag{4}$$

In constraint (3), $x_{ij} = 0$ indicates that facility j does not serve customer i and $x_{ij} = 1$ indicates that facility j serves customer i. In constraint (4), $p_j = 0$ or 1 indicates that facility j is closed or open respectively. Constraint (1) states that each customer must be and must only be served by one facility. Constraint (2) states that a facility can serve customers only if it is open. The objective of the UFLP is to minimize the sum of the customer serving costs and the facility opening costs.

The UFLP is also called the uncapacitated warehouse location problem or the simple plant location problem in the literature. Since the UFLP is an NP-hard problem (Cornuéjols et.al., 1990), exact algorithms ((Körkel, 1989) is an example) in general solve only small instances. For larger instances, approximation algorithms, heuristics, and metaheuristics have been proposed in the literature to solve this problem. Hoefer (2002) presents an experimental comparison of five state-of-the-art heuristics: JMS, an approximation algorithm (Jain

et.al., 2002); MYZ, also an approximation algorithm (Mahdian et.al., 2002); swap-based local search (Arya et.al., 2001); tabu search (Michel andVan Hentenryck, 2003); and the volume algorithm (Barahona and Chudak, 1999). Hoefer concluded that based on experimental evidence, tabu search achieves best solution quality in a reasonable amount of time and is therefore the method of choice for practitioners. Although approximation algorithms are more interesting in theory, heuristics and metaheuristics often outperform them in practice. Therefore, developing heuristics or metaheuristics for the UFLP has attracted more attention from researchers.

(Kuehn and Hamburger, 1963) presented the first heuristic that consists of two phases. In the first phase, the ADD method is applied that starts with all facilities closed and keeps adding the facility that results in the maximum decrease in the total cost. The first phase ends when adding any more facility will not reduce the total cost. In the second phase, the swap method is applied in which an open facility and a closed facility is interchanged as long as such an interchange reduces the total cost. Another greedy heuristic called the DROP method was also proposed by researchers (Nemhauser et.al., 1978). The DROP method starts with all facilities open, keeps closing the facility that results in the maximum decrease in the total cost, and stops if closing any more facility will not reduce the total cost. (Erlenkotter, 1978) proposed a dual approach for the UFLP. It is an exact algorithm but it can also be used as a heuristic.

In recent years, some metaheuristics were proposed for the UFLP. In general, metaheuristic methods spend more computation time and obtain better solution quality than heuristic methods do. These metaheuristics include genetic algorithms (Kratica et.al. 2001), tabu search (Ghosh, 2003)(Michel and Van Hentenryck, 2003)(Sun, 2006), and path relinking (Resende and Werneck, 2006). In particular, the most recent hybrid multistart heuristic proposed by Resende and Werneck and the tabu search proposed by Sun had made significant improvements in solving the benchmark instances. The hybrid multistart heuristic (Resende and Werneck, 2006) builds an elite set and applies path-relinking repeatedly to search for good solutions. Sun's tabu search (Sun, 2006) divides its search process into the short term memory process, the medium term memory process and the long term memory process. A move in this tabu search is to open or to close a facility. Sun also designed an efficient method to update, rather than re-compute, the net cost change of a move. In spite of all these improvements there is still room for making progress. In this paper, we present the Multistart Drop-Add-Swap heuristic (MDAS) for the UFLP.

The MDAS starts with a set of randomly generated initial solutions. A solution is represented by a sequence of n bits. The value of the ith bit is 0 (1) if the ith facility is close (open). For each solution, the following process is iterated a predetermined number of times. The MDAS firstly keeps closing the facility that results in the maximum decrease of the total cost until no facility can be closed to reduce the total cost. Then, the MDAS tries to find either opening a facility or interchanging an open facility with a closed facility will result in the maximum reduction of the total cost, and it will then do the corresponding opening or interchanging.

The rest of the paper is organized as follows. In Section 2, the proposed MDAS is described. Experimental results and some discussions are listed in Section 3. Finally, conclusions are given in Section 4.

2 MDAS HEURISTIC

In this section, we introduce the proposed Multistart Drop-Add-Swap heuristic (MDAS) for the UFLP. The Add method and the Swap method had been used to solve the UFLP by (Kuehn and Hamburger, 1963). Their heuristic consists of an Add method phase followed by a Swap method phase. The Add method phase starts with all facilities closed and keeps opening the facility that results in the maximum reduction in the total cost, and stops if opening any more facility will no longer reduce the total cost. In the Swap method phase, an open facility and a closed facility are interchanged as long as such an interchange decreases the total cost. The Drop method had also been proposed to solve the UFLP in previous studies. (Cornuéjols et.al., 1977) presented a heuristic that consists of a single phase of the Drop method. The Drop method starts with all facilities open and keeps closing the facility that results in the maximum reduction in the total cost, and stops if closing any more facility will no longer decreases the total cost. Unlike the above mentioned heuristics, the proposed DAS heuristic contains iterations of Drop or Add or Swap operation with one operation in each iteration. In each iteration, the Drop operation is first examined and if applying the Drop operation can reduce the total cost, it is applied; Otherwise, the Add operation or the Swap

operation whichever can reduce the total cost most, is applied. Moreover, multistart is utilized to enhance the global search capability of the proposed DAS heuristic because in multistart DAS (MDAS), search will begin with initial solutions distributed over the whole solution space.

(Sun, 2006) designed an efficient method to calculate the net cost change of an Add operation and a Drop operation, as denoted by equations (5) and (6) respectively.

$$\Delta^{+}O_j = \sum_{i \in X_j} \left(c_{id_i^1} - c_{ij} \right) - f_j \tag{5}$$

$$\Delta^{-}O_j = f_j - \sum_{i \in Y_j} \left(c_{id_i^2} - c_{ij} \right) \tag{6}$$

For a solution s, we define two sets of facilities: S_c represents the set of all closed facilities and S_o represents the set of all open facilities. Furthermore, for each customer i, let d_i^1 be the open facility (i.e. $d_i^1 \in S_o$) that is closest to i , and let d_i^2 be the open facility (i.e. $d_i^2 \in S_o$) that is second closest to i. X_j denotes the set of customers that will be supplied by facility j when facility j is changed from closed to open. Y_j denotes the set of customers that were originally supplied by facility j when facility j is changed from open to closed. With the above definitions, equation (5) represents the net cost reduction of opening facility j and equation (6) represents the net cost reduction of closing facility j. Obviously, if $\Delta^{+}O_j$ ($\Delta^{-}O_j$) is positive, opening facility j (closing facility j) will reduce the total cost. Since Sun utilized only the Add operation and the Drop operation, and we use three operations: Add, Drop and Swap, we further design the following three equations for net cost reduction calculation of a Swap operation (Open facility j and close facility k).

$$\overline{Z}_{jk} = \Delta^{+}O_j + f_k \tag{7}$$

$$Q_{ij} = \max\left\{ c_{id_i^1} - c_{ij},\ c_{id_i^1} - c_{id_i^2} \right\}, i \in Y_k - X_j \tag{8}$$

$$Z_{jk} = \overline{Z}_{jk} + \sum_{i \in Y_k - X_j} Q_{ij} \tag{9}$$

Equation (7) denotes the cost reduction produced by opening facility j and the elimination of the opening cost of facility k because k will be closed. Equation (8) represents the cost reduction produced by the reallocation of the customers originally supplied by facility k. Note that those customers originally supplied by facility k and will now be supplied by the newly opened facility j have already been considered in equation (7). Finally, the net cost

```
DAS( s )
/*Initialization phase*/
1   Count ← 0;
2.  Calculate the saving cost of dropping facility k, ΔˉOk, for each facility k∈So
3.  Calculate the saving cost of adding facility j, Δ+Oj, for each facility j∈Sc
4.  Calculate the saving cost of swapping facility j and facility k, Zjk, and find the
    best swap target Ej for each facility j∈Sc
/*Search phase*/
5.  repeat
6.      Operator ← Ø;
7.      Update the saving cost of dropping facility k, ΔˉOk .
8.      D ← arg max{ ΔˉOj | j ∈ So }; /*Get the best dropping facility D*/
9.      if ΔˉOD>0 then Operator ← Drop ( D );
10.     else
11.         Update the saving cost of adding a facility, Δ+Oj .
12.         Â ← arg max{ Δ+Oj | j ∈ Sc }; /*Get the best adding facility Â */
13.         Update the saving cost of swapping facility j and facility k, Zjk and Ej .
14.         Â ← arg max{ Zjk | j ∈ Sc }; /*Get the best swapping facility Â */
15.         D̂ ← Eˆ ; /*Get the best swapping facility D̂ */
16.         if Δ+OÂ>ZÂD̂ then
17.             if Δ+OÂ>0 then Operator ← Add( Â );
18.         else if ZÂD̂ >0 then Operator ← Swap( Â , D̂ );
19.     if Operator ≠ TabuOperator then
20.         Count ← Count + 1;
21.         Execute Operator;
22.         TabuOperator ← inverse of Operator ;
23.     else Operator ← Ø;
24.     if Count > n then Operator ← Ø;
25. until Operator=Ø
```

Figure 1: The DAS heuristic.

reduction of Swap(j, k), that is, opening j and closing k, is denoted by equation (9).

The pseudo code for the Drop-Add-Swap heuristic (DAS) is shown in Figure 1. In the pseudo code, Drop(D) denotes closing facility D, Add(A) denotes opening facility A, and Swap(A, D) denotes opening facility A and closing facility D simultaneously. As for the Multistart Drop-Add-Swap heuristic (MDAS), it will randomly generate a set of initial solutions first, and then it will apply the DAS heuristic to each initial solution to search for the optimum or near-optimal solution.

We now explain the DAS heuristic depicted in Figure 1. Given a solution s, for each open facility k, the saving cost of closing facility k will be calculated; for each closed facility j, the saving cost of opening facility j will be calculated; for each closed facility j, the saving cost of opening facility j and closing its best swapping target E_j will also be calculated (lines 2-4). After the initialization phase, begins the search phase. The search phase contains at most n iterations and in each iteration, at most one of the three operations (Drop, Add and Swap) will be applied. The algorithm terminates if the number of iterations is greater than n or the chosen operation is tabued or none of the three operations can reduce the total cost. In each iteration, the Drop operation is considered first. If the Drop operation can reduce the total cost, the Drop operation that reduces cost most will be applied (lines 7-9). Otherwise, the Add operation and the Swap operation will be considered simultaneously. Whichever of the Add operation and the Swap operation can reduce the total cost most

will be applied (lines 11-18). The inverse operation of the applied operation will be tabued in the next iteration (line 22). When applying any of Drop, Add or Swap operation, from equations (5)-(9), it is noted that only related information needs to be updated rather than all information needs to be recomputed. Therefore, the calculation of the net cost change of an operation is very time efficient. Moreover, by properly combining Drop, Add and Swap operations, the MDAS heuristic is very effective in solving the UFLP.

3 EXPERIMENTAL RESULTS AND DISCUSSIONS

Extensive experiments had been conducted to evaluate the performance of the proposed MDAS heuristic. A personal computer with Intel Core 2 Duo 2.33 GHz CPU was used as the platform to run the program. Although there were two processors, only one processor was used. The program was implemented using C++. In this section, experimental results are given and some discussions are made.

3.1 Benchmarks

Most benchmarks collected in the UflLib (Hoefer, 2002) website had been utilized to test the performance of the MDAS heuristic. These benchmarks are listed as follows.

ORLIB: This benchmark was proposed by (Beasley, 1993) and posted in OR-Library. It consists of 15 problems that are divided into four classes.

M*: These problems were presented by (Kratica et al., 2001). The benchmark contains six sets with problem size 100, 200, 300, 500, 1000 and 2000 respectively. Each set has five problems.

GR: The benchmark was proposed by (Galvao and Raggi, 1989). It consists of five sets with problem size 50, 70, 100, 150 and 200 respectively. Each set contains ten problems.

BK: This benchmark, presented by (Bilde and Krarup, 1977), contains 220 problems with size from 30×30 to 100×100. The distances between customers and facilities are drawn uniformly from [0, 1000].

FPP: This benchmark was artificially generated to make the problems harder to be solved. It was proposed by (Kochetov and Ivanenko, 2003) at the 5th Metaheuristics International Conference in 2003. The benchmark contains 40 problems and these problems are classified into two classes: FPP11 (with m = 133 and n = 11) and FPP17 (with m = 307 and n = 17).

GAP: This benchmark was also proposed by (Kochetov and Ivanenko, 2003) at the 5th Metaheuristics International Conference. It consists of three classes: GAPA, GAPB and GAPC. Each class contains 30 problems. Since these problems have larger duality gaps, they are more difficult to be solved by the dual-base method.

GHOSH: (Ghosh, 2003) presented this benchmark in 2003. The benchmark contains 90 problems. These problems are divided into two classes: symmetric and asymmetric. Each class is further divided into three sets with problem size 250, 500 and 750 respectively. The optimal solutions of some of these problems are still unknown.

MED: Originally, this benchmark was proposed by (Ahn et al., 1998) as a benchmark for the p-median problem. In 1999, (Barahona and Chudak, 1999) modified it to act as a benchmark for the UFLP. This benchmark consists of 18 problems with size 500, 1000, 1500, 2000, 2500 and 3000 respectively. For each size, there are three problems with the opening costs of facilities being set to $\sqrt{n}/10$, $\sqrt{n}/100$ and $\sqrt{n}/1000$, respectively.

3.2 Performance comparison of MDAS Heuristic and DROP Method

The comparison of performance of the DROP method, the DAS heuristic and the MDAS heuristic is shown in Table 1. The first column of Table 1 represents the problem name, the second column represents the size of the problem, where m is the number of customers and n is the number of facilities. The third column lists the costs of the optimal solutions. The fourth and the fifth column list the deviation from the optimal solution of the solution found by the DROP method and the DAS heuristic, respectively, with the initial solution in which all facilities were open.

The sixth column represents the average deviation (from the optimal solutions) of the solutions found by applying the MDAS heuristic ten times. In each time, ten randomly generated solutions were used as the initial solutions. The seventh column shows the number of times by which the optimal solution was found by the MDAS heuristic. Finally, the last column denotes the average CPU time used by the MDAS heuristic over ten times.

Table 1: Performance comparison of the DROP method, the DAS heuristic and the multistart DAS heuristic.

Problem	OPT	Drop	DAS	MDAS		
		Dev	Dev	AVG Dev	Hit	AVG Time
Cap71	932615.75	0.00	0.00	0.00	10	0.000
Cap72	977799.40	0.00	0.00	0.00	10	0.000
Cap73	1010641.4	0.00	0.00	0.00	10	0.000
Cap74	1034976.9	0.26	0.00	0.00	10	0.000
Cap101	796648.44	0.00	0.00	0.00	10	0.000
Cap102	854704.20	0.09	0.00	0.00	10	0.000
Cap103	893782.11	0.00	0.00	0.00	10	0.000
Cap104	928941.75	0.61	0.00	0.00	10	0.000
Cap131	793439.56	0.00	0.00	0.00	10	0.000
Cap132	851495.32	0.09	0.00	0.00	10	0.000
Cap133	893076.71	0.08	0.08	0.00	10	0.000
Cap134	928941.75	0.61	0.00	0.00	10	0.000
CapA	17156454.	6.95	0.00	0.00	10	0.084
CapB	12979071.	1.04	0.00	0.00	10	0.100
CapC	11505594.	1.24	0.26	0.00	10	0.109
MO1	1156.91	0.53	0.00	0.00	10	0.09
MO2	1227.67	0.00	0.00	0.00	10	0.08
MO3	1286.37	0.94	0.14	0.00	10	0.09
MO4	1177.88	0.16	0.00	0.00	10	0.09
MO5	1147.60	0.00	0.00	0.00	10	0.08
MP1	2460.10	0.00	0.00	0.00	10	0.041
MP2	2419.32	0.00	0.00	0.00	10	0.042
MP3	2498.15	0.00	0.00	0.00	10	0.042
MP4	2633.56	0.45	0.00	0.00	10	0.044
MP5	2290.16	0.00	0.00	0.00	10	0.044
MQ1	3591.27	0.00	0.00	0.00	10	0.109
MQ2	3543.66	0.04	0.00	0.00	10	0.109
MQ3	3476.81	0.85	0.00	0.00	10	0.120
MQ4	3742.47	0.00	0.00	0.00	10	0.111
MQ5	3751.33	0.00	0.00	0.00	10	0.113
MR1	2349.86	0.00	0.00	0.00	10	0.362
MR2	2344.76	0.00	0.00	0.00	10	0.414
MR3	2183.24	0.00	0.00	0.00	10	0.400
MR4	2433.11	0.00	0.00	0.00	10	0.377
MR5	2344.35	0.00	0.00	0.00	10	0.372
MS1	4378.63	0.00	0.00	0.00	10	2.489
MS2	4658.35	0.05	0.00	0.00	10	2.586
MS3	4659.16	0.67	0.00	0.00	10	2.858
MS4	4536.00	0.00	0.00	0.00	10	2.427
MS5	4888.91	0.00	0.00	0.00	10	2.172
MT1	9176.51	0.00	0.00	0.00	10	14.986
MT2	9618.85	0.00	0.00	0.00	10	13.878
MT3	8781.11	0.00	0.00	0.00	10	14.654
MT4	9225.49	0.00	0.00	0.00	10	14.855
MT5	9540.67	0.00	0.00	0.00	10	13.775

From Table 1, it is noted that there are 17 problems on which the DROP method cannot find optimal solutions, but there are only 3 problems on which the DAS heuristic cannot find optimal solutions. Hence, the DAS heuristic is superior to the DROP method. Furthermore, when ten randomly generated solutions were used as the initial solutions instead of using only the solution with all facilities opened, the MDAS heuristic could find optimal solutions for all these 45 problems and for all ten times of experiments.

Therefore, the MDAS heuristic enhances the global search ability of the DAS heuristic and improves the performance significantly. By examining the solutions of the seventeen problems on which the DROP method could not find optimal solutions, some of the solutions were very close to the optimal solutions, but the optimal solutions could not be reached from these solutions by applying the Drop operation. The three problems on which the DAS heuristic could not find optimal solutions had a similar situation. The solutions found by the DAS heuristic for these three problems and the optimal solutions of these three problems are listed in Table 2. It is noted that two swap operations (Swap(6, 11) and Swap(25, 15)) executed simultaneously are needed to transform the solution found by the DAS heuristic to the optimal solution for Cap133. Similarly, more than two operations executed simultaneously are needed to transform the solutions found by the DAS heuristic to the optimal solutions for the other two problems. But when the DAS heuristic started with multiple randomly generated initial solutions, though still being restricted to executed only one operation in each iteration, the multi-start DAS (MDAS) could find all optimal solutions of all these forty-five problems as shown in Table 1.

Table 2: The optimal solutions and the solutions found by the DAS heuristic (all open).

Instance	Source	Solution
Cap133	OPT	6, 23, 25, 27, 34, 45, 46, 49
	DAS(All Open)	11, 15, 23, 27, 34, 45, 46, 49
Capc	OPT	6, 14, 24, 35, 53, 70, 79, 81, 89
	DAS(All Open)	6, 9, 14, 24, 35, 48, 66, 70, 79
MO3	OPT	6, 39, 48
	DAS(All Open)	7, 38, 39, 100

Table 3: Performance comparison of the MDAS heuristic with two other methods on six benchmarks.

Class	Method	ORLIB		M*		GR		BK		FPP		GAP	
		Avg D	Time	Avg D	Time	Avg D	Time	Avg D	Time	Avg D	Time	Avg D	Time
Hybrid	Iteration=8, Elite=5	**0.000**	0.05	0.004	2.19	**0.000**	0.09	0.028	0.09	69.370	1.63	9.573	0.348
	Iteration=32, Elite=10	**0.000**	0.17	**0.000**	7.86	**0.000**	0.32	0.002	0.28	33.375	7.66	5.953	1.64
	Iteration=2048, Elite=80	—	—	—	—	—	—	—	—	**0.000**	330.9	0.820	92.09
Simple Tabu	500 Non-improving	0.028	0.16	0.011	1.75	0.100	0.16	0.071	0.16	95.711	0.65	15.901	0.26
	64000 Non-improving	—	—	—	—	—	—	—	—	71.150	52.08	6.350	22.43
MDAS	10 Random Seed	**0.000**	0.02	**0.000**	0.92	**0.000**	0.02	0.032	0.00	46.948	0.04	13.783	0.01
	50 Random Seed	—	—	—	—	—	—	0.001	0.02	3.963	0.21	8.661	0.03
	100 Random Seed	—	—	—	—	—	—	**0.000**	0.03	0.298	0.43	7.063	0.07
	1000 Random Seed	—	—	—	—	—	—	—	—	0.012	4.46	2.930	0.74
	4000 Random Seed	—	—	—	—	—	—	—	—	0.003	17.53	1.217	3.42
	16000 Random Seed	—	—	—	—	—	—	—	—	**0.000**	67.36	**0.559**	20.30

3.3 Performance Comparison of MDAS and the Other Two Methods

In this subsection, the performance of the MDAS is compared with those of the hybrid multistart heuristic and the simple tabu search on six benchmarks. This comparison is depicted in Table 3. The performance data of the hybrid multistart heuristic and the simple tabu search were taken from (Resende and Werneck, 2006). The six benchmarks considered are ORLIB, M*, GR, BK, FPP and GAP. The computer used by the MDAS was a personal computer with Intel core2 Duo 2.33GHz CPU (only one processor was used).

The computer used by the other two methods was the SGI challenge with 28 196-MHz MIPS R10000 CPU (only one processor was used). In this experiment, the MDAS was run 50 times on each problem and the average was taken. In Table 3, AvgD (in %) is defined as follows: AvgD=[(best solution found by the method - best known upper bound)/best known upper bound]*100.

Where the best known upper bound is taken from (Hoefer, 2002). AvgT (in seconds) denotes the average CPU time. It is noted that with just 10 seeds (i.e., 10 randomly generated initial solutions), the MDAS can find all the best solutions in all 50 times (i.e., AvgD=0) on ORLIB, M* and GR benchmarks. Also, the CPU time needed is much less than those needed by the other two methods. For BK and FPP benchmarks, the MDAS can also find all the best solutions in all 50 times with 100 and 16000 seeds, respectively. In solving these two benchmarks, the CPU time need by the MDAS is also much less than those needed by the other two methods. As for the benchmark GAP, the MDAS found the best solution of each problem in this benchmark within the 50 times of execution, but it could not find all best solutions in any single time of execution. From Table 3, it is observed that the MDAS outperforms the other two methods in both the solution quality and the computation time.

3.4 Performance Comparison of MDAS and the other Three Methods

In this subsection, the performance of the MDAS is compared with those of Ghosh's method (Ghosh, 1999), the hybrid multistart heuristic (Resende and Werneck, 2006), and Sun's tabu search (Sun, 2006) on the GHOSH benchmark. There are totally 90 problems in this benchmark that are divided according to size and type into 18 sets with five problems in each set. The performance comparison is shown in Table 4, within which the previous best known solutions are taken from (Sun, 2006). Both the data of the hybrid multistart heuristic and the data of our MDAS are the average of 50 runs. It is noted in Table 4 that Ghosh's method achieves one (out of eighteen) previous best known solution, the hybrid multistart heuristic achieves four (out of eighteen) previous best solutions, Sun's tabu search achieves fourteen (out of eighteen) previous best solutions, and our MDAS achieves thirteen (out of eighteen) previous best known solutions. Moreover, the MDAS found solutions of the other four sets that are better than the previous best known solutions (the italics solutions in Table 4). The last row of

Table 4: Performance comparison of the MDAS heuristic and three other methods on benchmark GHOSH.

Instance			Previous best known	Ghost		Hybrid		Tabu Search		MDAS(1000Seed)	
Class	Size	Type	Value	Best	Time	Best	Time	Best	Time	Best	Time
A	250	Sym	257805.0	257832.6	18.256	257807.9	5.3	257805.0	2.828	***257804.0***	7.36
		Asym	257917.8	257978.4	18.060	257922.1	5.7	**257917.8**	2.618	**257917.8**	7.32
	500	Sym	511180.4	511383.6	213.316	511203.0	43.5	**511180.4**	15.616	511181.2	35.87
		Asym	511140.0	511251.6	207.070	511147.4	40.3	511140.0	13.760	***511136.4***	35.97
	750	Sym	763693.4	763831.2	824.288	763713.9	112.6	763693.4	39.812	***763684.8***	93.68
		Asym	763717.0	763840.4	843.206	763741.0	117.5	763717.0	39.650	***763716.4***	92.60
B	250	Sym	276035.2	276185.2	6.470	**276035.2**	8.0	**276035.2**	5.628	**276035.2**	5.88
		Asym	276053.2	276184.2	6.402	276053.6	8.2	**276053.2**	5.790	**276053.2**	5.92
	500	Sym	537912.0	538480.4	71.394	537919.1	52.6	**537912.0**	31.432	**537912.0**	26.72
		Asym	537847.6	538144.0	79.192	537868.2	52.2	**537847.6**	34.748	**537847.6**	26.73
	750	Sym	796571.8	796919.0	409.372	796593.7	126.3	**796571.8**	93.352	**796571.8**	68.83
		Asym	796374.4	796754.2	395.958	796393.5	127.1	**796374.4**	95.430	**796374.4**	64.71
C	250	Sym	333671.6	**333671.6**	17.322	**333671.6**	8.3	**333671.6**	9.878	**333671.6**	5.73
		Asym	332897.2	333058.4	24.730	**332897.2**	7.4	**332897.2**	9.196	**332897.2**	5.67
	500	Sym	621059.2	621107.2	146.482	**621059.2**	50.8	**621059.2**	71.106	**621059.2**	24.18
		Asym	621463.8	621881.8	134.76	621475.2	57.4	**621463.8**	72.064	**621463.8**	26.60
	750	Sym	900158.6	900785.2	347.414	900183.8	130.3	**900158.6**	229.914	**900158.6**	56.31
		Asym	900193.2	900349.8	499.738	900198.6	136.5	**900193.2**	236.902	**900193.2**	56.95
Average				555535.489	236.847	555493.567	60.556	555316.189	56.096	**555315.500**	34.730

Table 4 lists the average values over the eighteen sets, and from the average values, it is noted that the MDAS outperforms the other three methods.

3.5 Performance Comparison of MDAS and Hybrid Multistart Heuristic on MED

The performance of the MDAS is compared with that of the hybrid multistart heuristic on the benchmark MED, and this comparison is shown in Table 5. The MED benchmark contains 18 problems with size 500, 1000, 1500, 2000, 2500 and 3000, respectively. For each size *n*, there are three problems with the opening costs of facilities being set to $\sqrt{n}/10$, $\sqrt{n}/100$, and $\sqrt{n}/1000$, respectively.

In this experiment, the MDAS heuristic was run 50 times with 1000 seeds. It is noted in Table 5 that the MDAS heuristic outperforms the hybrid multistart heuristic on the first group in which the facility's open cost is $\sqrt{n}/10$. But the solution quality of the hybrid multistart heuristic is better than that of the MDAS heuristic on the second and the third groups in which the facility's open costs are $\sqrt{n}/100$ and $\sqrt{n}/1000$, but the computation time of the latter is less than that of the former.

Observing the above mentioned experimental results, we conclude the following:

(1) The DAS heuristic that utilizes the Drop operation, the Add operation and the Swap operation performs better than the DROP method that utilizes only the Drop operation.

Table 5: Performance comparison of the MDAS heuristic with the hybrid multistart heuristic on benchmark MED.

instance	Hybrid		MDAS(1000Seed)	
	Avg	Time	Avg	Time
500-10	**798577.0**	33.2	**798577.0**	28.0
1000-10	1434185.4	173.9	**1434171.0**	130.7
1500-10	2001121.7	347.8	**2000854.14**	331.1
2000-10	2558120.8	717.5	2558121.5	687.4
2500-10	3100224.7	1419.5	**3100174.5**	1116.7
3000-10	3570818.8	1621.1	3570820.75	1667.0
500-100	**326805.4**	32.9	326922.375	25.6
1000-100	**607880.4**	148.8	607992.563	106.5
1500-100	**866493.2**	378.7	867149.688	293.6
2000-100	**1122861.9**	650.8	1123936.5	562.3
2500-100	**1347577.6**	1128.2	1348713.25	870.5
3000-100	**1602530.9**	1977.6	1605083.63	1349.5
500-1000	**99196.0**	23.6	**99196.0**	22.4
1000-	**220560.9**	141.7	220626.563	84.4
1500-	**334973.2**	387.2	335400.813	218.7
2000-	**437690.7**	760	438263.0	425.9
2500-	**534426.6**	1309.4	535134.938	675.3
3000-	**643541.8**	2081.4	644376.25	1017.3

(2) The MDAS heuristic enhances the global search capability of the DAS heuristic and the performance of the MDAS heuristic will be improved as the number of seeds increases.
(3) Although the MDAS heuristic is just a heuristic, its performance is better than some state-of-the-art metaheuristics. (Table 3 and 4)

The fact that the hybrid multistart heuristic outperforms the MDAS heuristic on the second and the third groups of the MED benchmark (Table 5) indicates that the global search ability of the MDAS heuristic is still not good enough for searching a large solution space.

4 CONCLUSIONS

In this paper, the DAS heuristic and the multistart DAS (MDAS) heuristic are proposed to solve the UFLP. The DAS heuristic utilizes three operations: the Drop operation, the Add operation, and the Swap operation. And the MDAS heuristic enhances the global search capability of the DAS heuristic by applying it multiple times with different initial solutions (seeds). Experimental results reveal that the MDAS heuristic outperforms other state-of-the-art heuristics on most of the benchmarks. But the global search ability of the MDAS heuristic is still not good enough for searching a large solution space, therefore, in future studies, we will try to combine a global search scheme with the DAS heuristic to improve the performance. Also, we plan to investigate the possibility of applying the proposed heuristic to other combinatorial problems.

REFERENCES

Ahn, S., Cooper, C., Cornuéjols, G., Frieze, A.M., 1998. Probabilistic analysis of a relaxation for the k-median problem. Mathematics of Operations Research, (13) 1-31.

Arya, V., Garg, N., Khandekar, R., Meyerson, A., 2001. Munagala K, Pandit V. Local Search heuristics for k-median and facility location problems. ACM Symposium on Theory of Computing, 21-29.

Barahona, F., Chudak, F., 1999. Near-optimal solutions to large scale facility location problems. Technical Report RC21606, IBM, Yorktown Heights, NY, USA.

Beasley, J.E., 1993. Lagrangean heuristics for location problems. European Journal of Operational Research, (65) 383-399.

Bilde, O., Krarup, J., 1977. Sharp lower bounds and efficient algorithms for the simple plant location problem. Annals of Discrete Mathematics, (1) 79-97.

Cornuéjols, G., Nemhauser, G.L., Wolsey, L.A., 1990. The uncapacitated facility location problem. in: P.B. Mirchandani, R.L. Francis (Eds.), Discrete Location Theory, Wiley-Interscience, New York, 119-171.

Erlenkotter, D., 1978. A dual-based procedure for uncapacitated facility location. Operations Research, (26) 992-1009.

Galvao, R.D., Raggi, L.A., 1989. A method for solving to optimality uncapacitated facility location problems. Annals of Operations Research, (18) 225-244.

Ghosh, D., 2003. Neighborhood search heuristics for the uncapacitated facility location problem. European Journal of Operational Research, (150) 150-162.

Hoefer, M., 2002. Performance of heuristic and approximation algorithms for the uncapacitated facility location problem. Research Report MPI-I-2002-1-005, Max-Planck-Institutfür Informatik.

Hoefer, M., 2002. UflLib, http://www.mpi-sb.mpg.de/units/agl/projects/benhmarks/UflLib.

Jain, K., Mahdian, M., Saberi, A., 2002. A new greedy approach for facility location problems, in: Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC), ACM Press, 731-740.

Kochetov, Y., Ivanenko, D., 2003. Computationally difficult instances for the uncapacitated facility location problem. in: Proceedings of the 5th Metaheuristics International Conference (MIC), 41:1-41:6.

Körkel, M., 1989. On the exact solution of large-scale simple plant location problems, European Journal of Operational Research, (39) 157-173.

Kratica, J., Tosic, D., Fillipovic, V., Ljubic, I., 2001 Solving the simple plant location problem by genetic algorithm. RAIRO Operations Research, (35) 127-142.

Kuehn, A.A., Hamburger, M.J., 1963 A heuristic program for locating warehouses. Management Science (9) 643-666.

Mahdian, M., Ye, Y., Zhang, J., 2002. Improved approximation algorithms for metric facility location problems, in: Proceedings of the 5th International Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX), volume 2462 of Lecture Notes in Computer Science, Springer-Verlag, 229-242.

Michel, L., Van Hentenryck, P., 2003. A simple tabu search for warehouse locatiohn. European Journal of Operational Research, (157) 576-591.

Nemhauser, G.L., Wolsey, L.A., Fisher, L.M., 1978 An analysis of approximations for maximizing submodular set functions, I. Mathematical Programming, (14) 265-94.

Resende, M.G.C., Werneck, R.F., 2006. A hybrid multistart heuristic for the uncapacitated facility location problem. European Journal of Operational Research, (174) 54-68.

Sun, M., 2006. Solving the uncapacitated facility location problem using tabu search. Computers & Operations Research, (33) 2563-2589.

PROVIDING SPATIAL INTEGRITY FOR DISTRIBUTED UNMANNED SYSTEMS

Peter Simon Sapaty
Institute of Mathematical Machines and Systems, National Academy of Sciences, Glushkova Ave 42, 03187 Kiev, Ukraine
sapaty@immsp.kiev.ua

Keywords: Unmanned systems, Distributed scenario language, Networked interpretation, System integrity, robotic swarms, Reconnaissance, Camp security, Convoys, Explosive ordnance disposal, Gestalt, World super-machine.

Abstract: Due to the increased complexity of tasks delegated to unmanned systems, their collective use is becoming of paramount importance for performing any reasonable jobs. An approach is offered where group behaviors are accomplished automatically rather than set up manually, as usual. Missions in the Distributed Scenario Language (DSL) can be executed jointly by communicating interpreters in system units. Scenarios like reconnaissance, camp security, convoy, mule, and explosive ordnance disposal in DSL, oriented on different numbers of cooperating units, are demonstrated. The approach allows us to effectively manage any teams, from human to robotic, and from homogeneous to heterogeneous, regardless of the number of components in them. A variety of other applications of the technology are outlined too, already researched or prospective, also its relation to the gestalt philosophy, where super-summative whole dominates over system parts, defining their sense and even existence, rather than vice versa. The paradigm discussed may also represent a distributed dynamic world super-machine operating in parallel with both information and physical matter.

1 INTRODUCTION

With the world dynamics increasing due to global warming, numerous natural and manmade disasters, military conflicts, and international terrorism, using unmanned (ground, sea, underwater, and air) systems can alleviate many problems and save lives in hazardous environments. Because of the complexity of tasks delegated to unmanned solutions and still insufficient capabilities of existing robotic vehicles, their simultaneous, collective use may be of paramount importance to perform any reasonable jobs. Operating together, the unmanned groups, often called *swarms*, can fulfill the required objectives despite possible runtime damages to individual units.

We are offering a novel approach to organization of unmanned systems, oriented from the very beginning on parallel solutions in physical spaces, with swarm behaviors resulting naturally and accomplished automatically, rather than programmed manually. This approach, symbolically called "overoperability" from the previous publications (Sapaty, 2002, 2005), allows us to create, modify, analyze, process, and manage any distributed systems, establishing local and global dominance over them.

Within the overoperability philosophy, an integral mission scenario, written in a special high-level scenario language (Sapaty, 1999, 2005) and reflecting semantics of what to be done in a distributed space rather than details of implementation, is executed in a parallel and cooperative manner by dynamically networked unmanned units. During the scenario evolution, any operations can be accomplished in the world, along with the needed movement of code, equipment and "doers" (both artificial and biological), as well as creation and maintenance of physical and virtual infrastructures supporting the missions.

This paper is essentially inspired by the European Land Robotic Trial (M-ELROB, 2008) in which the author participated. It was conducted to provide trials as close as possible to operational scenarios for UGVs/UAVs with focus on short-term realizable robot systems. The day and night trials were organized within the following five main scenarios: non-urban reconnaissance, camp security, transport convoy, transport mule, and explosive ordnance disposal. Only a limited number of robotic units were engaged in every scenario, just one or

two, whereas every scenario could potentially be executed with much higher efficiency if using robotic teams with many units, which cooperate with each other.

The paper also reflects activity on the project started under the sponsorship of Alexander von Humboldt Foundation (AvH) in Germany. One of its aims is formalization of known mission scenarios in such a way that they could be performed by any available numbers of robotic vehicles, with the management burden effectively shifted to self-organized robotic teams -- thus relieving human operators from traditional routines and allowing them concentrate on mission goals and global efficiency instead.

2 DISTRIBUTED SCENARIO LANGUAGE (DSL)

The approach described here is based on the Distributed Scenario Language (DSL), which allows us to set what to do in a distributed world on a semantic level, abstracting from details of how to do this and with which resources, delegating these to the intelligent automatic interpretation. Being a universal programming language with advanced parallel and distributed capabilities, DSL can also describe tasks and behaviors on any levels, if needed. The language can be used by humans who should follow its instructions individually or collectively, or can be directly executed by robots and their teams. Any mixed human-robotic organizations can be managed in DSL too.

2.1 The World

The world DSL operates with can be virtual, physical, or combined.

- *Virtual World (VW)* is discrete and consists of nodes and links connecting these nodes. Any information can be associated with both nodes and links in the form of their names (contents). Nodes have unique addresses in VW, whereas their names (same as names of links) may repeat throughout the VW. Nodes can be accessed directly, globally, by their names or addresses, or locally, from each other, via the (named) links, whereas links can be accessed only locally--from the adjacent nodes. A variety of broadcasting possibilities are available in the VW, both in a global and local way, for example, from outside to all nodes, from a node directly to all other nodes, or from a node to all neighboring nodes via the selected or all adjacent links.
- *Physical World (PW)* is continuous. Any point in it can be identified and accessed by the coordinates expressed in a certain coordinate system, also with certain precision. Staying in a PW point, you can lift local physical parameters from the world and, possibly, also change them, impacting the world locally too.
- *Virtual-Physical World (VPW)* is the one where VW nodes additionally associate with certain coordinates in the PW. VPW is discrete on a snapshot, but the nodes can change their physical coordinates overtime. The VPW nodes can be globally accessed by their names, addresses, or physical coordinates (for the latter, more correctly: by coordinates of the expected center and a radius of the region, due to limited precision of the coordinates). Also locally, via links--same as for the pure VW. In addition to the broadcasting capabilities of VW, nodes in VPW can also be massively accessed/entered by identifying a probable region in PW where they are expected to exist--by the region's center and a range (radius) from this center, where the latter may be of any value.

2.2 High-Level Scenarios

The world, as described above, is navigated and processed in a parallel and distributed way by high-level DSL scenarios having the following main features.

General Features:

- A DSL scenario (or program, in a conventional notation) describes development of activities in a distributed world as parallel transitions between sets of progress points, or *props*.
- Starting from a prop, a program action may result in one or more new props, or remain in the same prop.
- Each prop has a resultant *value* and a resultant *state*.
- Different actions (whatever complex they might be), starting from the same prop, may evolve independently or interdependently, and sequentially or in parallel, each contributing to the resultant set of props on this group of actions.
- Actions may also succeed each other in the space of props, with new actions applied in parallel from all props reached by the previous actions, resulting altogether in the integrated set of props on all these applications.
- Elementary operations can be defined on the

values of props reached by other actions (the latter of any complexity), leading to the resultant prop with associated value (which may be multiple) and resultant state.

- The scenarios can form new or remove existing nodes and links in the distributed VW or VPW, allowing us to create, modify, and process any graph-based infrastructures in these worlds.

Association with World Nodes/Points:

- Any prop can be associated with a *node* in VW or a *position* in PW, or both, like in the case of VPW.
- A prop can also be linked separately with VW nodes and PW positions, allowing us to operate with the two worlds independently.
- Any number of props can be associated simultaneously with same points of the worlds.
- Staying with nodes/positions in the worlds, a prop allows us to directly access local data in these points, both virtual (information) and physical (matter).

Different Types of Variables:

- *Heritable variables* – these are starting in a prop and serving all subsequent props, which can share them in both read & write operations.
- *Frontal Variables* – are an individual and exclusive prop's property (not shared with other props), being transferred between the consecutive props, and replicated if from a single prop a number of props emerge.
- *Environmental Variables* – are accessing different elements of physical and virtual words when navigating them, also a variety of parameters of the internal world of DSL interpreter.
- *Nodal Variables* – allow us to attach an individual temporary property to VW and VPW nodes; they can be accessed and shared by any props associated with these nodes.
- Different types of variables, especially when used together, allow us to create efficient spatial algorithms which work *in between* components of distributed systems rather than *in* them.

Hierarchical Control:

- DSL scenarios can use a variety of spatial control rules, allowing us to assess local and remote states, make local and global decisions, and invoke or skip subsequent and terminate current operations, on the results of these decisions.
- Nested control infrastructures, embracing the whole scenario, provide interdependent local and global decisions associated with proper points of the worlds.

2.3 The Language Syntax

DSL has a recursive syntax shown below together with names of its main constructs (where square brackets are for an optional construct, braces mean construct repetition with a delimiter at the right, and vertical bar separates alternatives).

wave → *constant* | *variable* | [*rule*] ({ *wave* , })
constant → *number* | *string* | *special*
variable → *identifier* | *reserved*
rule → *expand* | *transfer* | *modify* | *branch* | *advance* | *repeat* | *grant* | *echo* | *arithmetic* | *structural* | *assign* | *compare* | *timing* | *type* | *usage* | *identifier* | *wave*
special → abort | thru | done | fail | any | random | all | out | in | infinite | nil | empty | first | last | andom | virtual | physical | combined | neighbors | global | local | direct
reserved → N{*alphameric*} | H{*alphameric*} | F{*alphameric*} | TYPE | QUALITIES | NAME | ADDRESS | PLACE | WHERE | BACK | PREVIOUS | LINK | DIRECTION | ORDER | WHEN | TIME | SPEED | STATE | VALUE | COLOR | RESOURCES | DOER | USER | START
expand → hop | move | create | linkup
transfer → run | call | output
modify → split | partition | select | replicate | *integer*
branch → par | sequence | if | while | or | par or | and | par and
advance → advance | sync advance
repeat → cycle | loop | sling | repeat | repeat sync
grant → free | release | quit | none | lift | stay | grasp
echo → rake | min | max | sort | sum | average | product | count | state
arithmetic → add | subtract | multiply | divide | degree
structural → separate | unite | concatenate | append | intersect | content | index | rand
assign → assign | assign peers
verify → equal | not equal | less | less equal | more | more equal | empty | nonempty | belong | not belong | inside | not inside
timing → sleep | remain
type →nodal | heritable | frontal | environmental | info | matter | number | string | wave
usage → address | name | place | center | range | time | speed | doer | node | link | unit

The DSL top level structure can also be expressed graphically, as in Fig. 1. The basic construct, *rule*, can represent any action or decision and can, for example, be as follows:

- Elementary arithmetic, string or logic operation.
- Hop in a physical, virtual, or combined space.
- Hierarchical fusion and return of (remote) data.
- Distributed control, both sequential and parallel.
- A variety of special contexts for navigation in space, influencing operations and decisions.
- Type or sense of a value, or its chosen usage, guiding automatic interpretation.

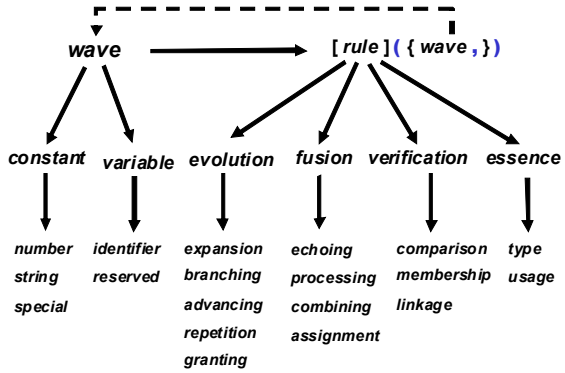


Figure 1: Recursive structure of DSL.

Different variants of this syntax and semantics had been implemented for previous DSL subsets (Sapaty, 1999, 2005), where conventional expression of operations and delimiters between program parts can be used too, say, for better readability and compactness. For example:

`add(3, 5, 7)` same as `3 + 5 + 7`

`advance(w1, w2, w3)` same as `w1; w2; w3`

where *w1* to *w3* may be arbitrary DSL programs (*waves*) themselves. The first example could have any programs instead of just numbers, each returning its (possibly, remote and multiple) results, as follows:

`add(w1, w2, w3)` same as `w1 + w2 + w3`

3 DISTRIBUTED INTERPRETER

A variety of options may be available for automatic interpretation of DSL scenarios – from fully centralized and sequential to fully distributed and parallel. Due to peculiar syntax and semantics, the language interpretation in distributed systems is transparent and straightforward. Some basic features of the DSL interpretation are as follows.

- Direct association of props with world points drastically simplifies bringing data from the points to scenarios or vice versa: scenarios or their parts to world points.
- Chained actions can self-navigate and match the world, while omitting used "heads" and forwarding remaining "tails" further.
- Independent actions can be launched in parallel, developing autonomously in parts of the world.
- The interpreter copy can be installed in internet hosts, mobile robots, laptops, mobile phones, smart sensors, or implanted into biological units.
- The interpreter can also be a human being, performing manually of what is for herself while passing other parts of the scenario to other human or electronic interpreters and establishing dynamic command and control infrastructures between them.
- Any other systems can be accessed via the networked interpreters, the latter forming a supervisory layer managed in DSL.
- The interpreter copies may be concealed inside the systems to be impacted, even without their knowledge (to work in hostile environments).
- The interpreters can also migrate in the worlds to be managed, collectively executing (mobile too) mission scenarios, resulting altogether in a flexible and ubiquitous system organization.
- The DSL interpreter consists of a number of specialized modules working in parallel and handling and sharing specific data structures, which are supporting both persistent virtual worlds and temporary hierarchical control mechanisms (Sapaty, 1999, 2005).
- The heart of the distributed interpreter is its *spatial track system* enabling hierarchical command and control and remote data & code access, also providing high integrity of emerging parallel and distributed solutions, achievable without central facilities.

In application to robotic communities, the approach allows us to convert any group of mobile robots into a goal-directed cooperative system by integrating copies of the DSL interpreter, presented as a universal control module U in Fig. 2, with traditional robotic functionalities, as in Kuhnert, Krödel, 2005. (The figure exhibits mobile robots which participated in M-ELROB 2008 trial).

Any mission scenario in DSL can start from any robot, covering and tasking the whole system (or its parts needed) at runtime and in parallel. Subordination between the units and dynamic command and control are established automatically, as a derivative of the mission scenario and current state of environment.

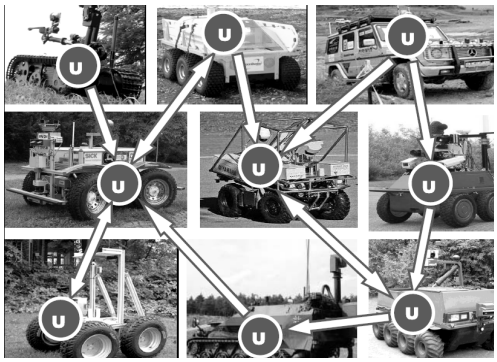


Figure 2: Heterogeneous robotic teaming using embedded DSL interpreters.

Due to fully interpretive nature of the technology, the scenarios can self-recover from any points, timely reacting on failures of robots. The whole group may remain fully functional and global-goal-oriented even in case of indiscriminate damages to individual units.

4 ELEMENTARY EXAMPLE

An elementary task to be programmed in DSL may look like follows:

Go to given physical locations of the disaster zone (represented in a proper system of coordinates by the three locations): (50.433, 30.633), (50.417, 30.490), and (50.467, 30.517). Evaluate damage in each location, then find and transmit the maximum destruction value on all locations, together with exact coordinates of the corresponding location, to a management center.

The corresponding program in DSL will be:

```
transmit(maximum(
  move((50.433, 30.633),
       (50.417, 30.490),
       (50.467, 30.517));
  attach(assess(damage),WHERE)))
```

This program reflects semantics of the task to be performed in a distributed space, regardless of possible equipment that can be used for this. The latter may, for example, be a set of sensors scattered in advance throughout the disaster zone, where hopping by coordinates may result in a wireless access of the sensors already present there, not necessarily moving into these points physically.

As another solution, the program may task mobile robots to move into these locations and perform the needed damage assessment upon reaching the destinations. We will be showing here this latter option, using three available robots R1, R2, and R3.

The possible starting position and initial scenario injection (let it be into R1) are shown in Fig. 3.

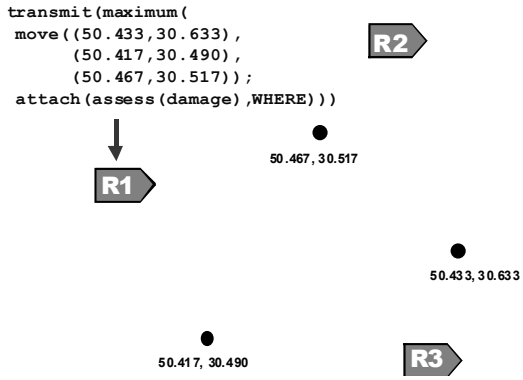


Figure 3: Initial scenario injection.

After the creation of a distributed interpretation infrastructure covering all three robots, R1 is partitioning the scenario, and modifying and tasking itself and the other two robots, as in Fig. 4.

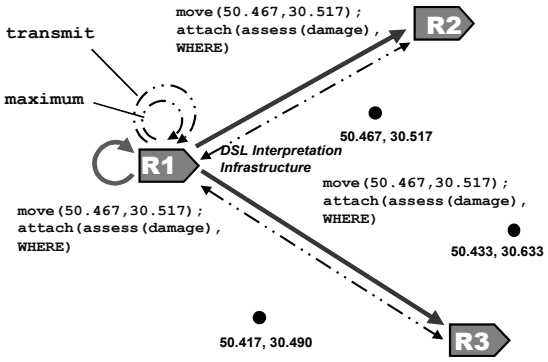


Figure 4: Parallel tasking of three robots.

All three robots then move independently to the locations optimally chosen for them, as in Fig. 5.

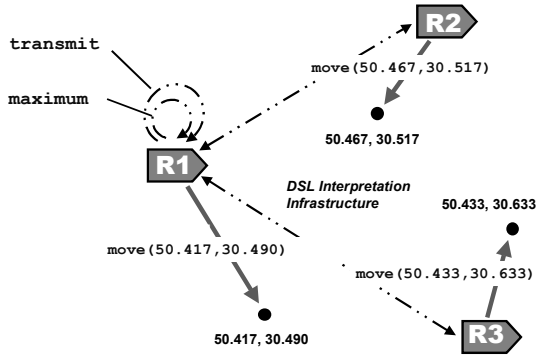


Figure 5: Simultaneous robot movement.

In each location reached independently by a corresponding robot, the damage assessment and exact coordinates return and attachment take place, as in Fig. 6.

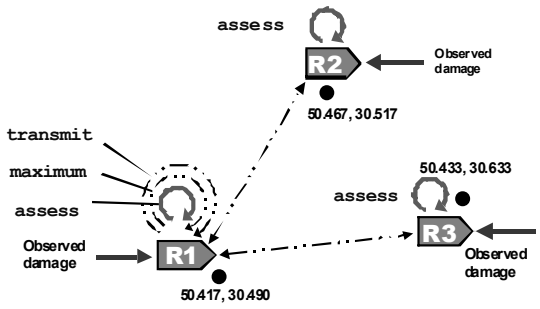


Figure 6: Simultaneous damage assessment.

And finally, R1, using rule `maximum`, finds global maximum damage value from those obtained in each of the three robots, and together with the corresponding location coordinates transmits it to the management center, as in Fig. 7.

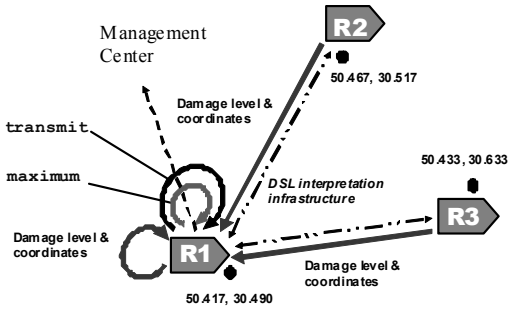


Figure 7: Merging data, finding global maximum.

As can be seen from the examples above, a semantic level scenario describing what to do in the distributed space, can be interpreted by robotic teams autonomously, and by different numbers of cooperating robots (we could use two or a single robot instead). The number of available robots can also vary at runtime, during the scenario evolution.

5 MORE ROBOTIC SCENARIOS

We will be using here the main scenarios that were the basis of the M-ELROB 2008 trial.

5.1 Non-urban Reconnaissance

For this scenario, it is supposed that a group of unknown vehicles is located in some distance in a non-urban area (defined, say, with the position of a center and area's radius), with security situation unclear there, so the reconnaissance should be done by robotic vehicles for not risking own personnel. The objective is to go to this target area and search for vehicles with specific characteristics. If found, they should be examined in detail, with their parameters collected and reported to the control station.

The general picture is shown in Fig. 8, where the reconnaissance facilities should first go to the target area (i.e. its center), observe the area by cameras/sensors to roughly locate most probable targets (by their size, for example). The next will be to move directly to these selected targets and sense & collect their detailed parameters, with sending the results to the control point where they are stored and analyzed.

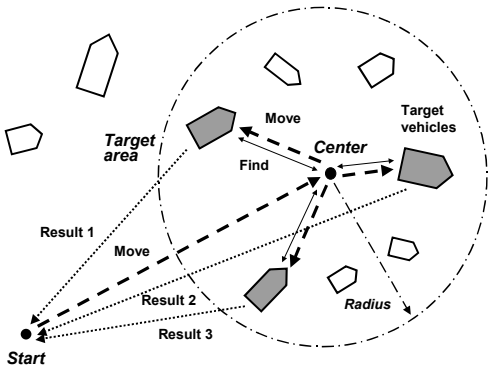


Figure 8: The reconnaissance scenario.

Parallel Solution. This, in DSL, may allow us to use as many reconnaissance vehicles as possible (a single one including), potentially involving individual vehicles for each target identified, for their detailed examination.

```
USER =(move(start); WHERE=center;
 Targets=recognize(radius,features);
 split(Targets); WHERE = VALUE;
 collect(size, type, speed))
```

Explicitly Sequential Solution. The following DSL program just details navigation and organization procedures to execute the reconnaissance scenario in a strictly sequential way, which may be useful for optimization of the use of a single vehicle only.

```
move(start); WHERE = center;
Targets=recognize(radius,features);
loop(WHERE = withdraw(Targets,1);
 Result &=collect(size,type,speed));
USER = Result
```

Avoiding Obstacles. The movement to the target area and inside it may be complicated due to presence of obstacles, as shown in Fig. 9. The following DSL program, for the move

from *Start* to *Center*, uses an external procedure `approach_or_stop` to detect obstacles and stop to avoid collision, and the procedure `suitable` to find next suitable waypoint on the way to the destination, from which the move should continue.

```
move(start);
 loop(approach_or_stop(center);
 WHERE != center;
 WHERE = suitable(depth,center));
```

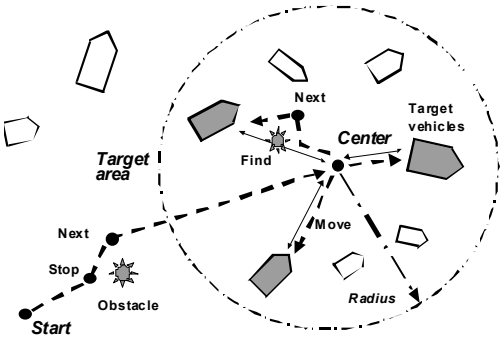


Figure 9: Avoiding obstacles.

5.2 Camp Security

For the camp security scenarios, a defined urban area has to be monitored (think military camp) and this should be executed by robotic vehicles too, to minimize risk to human personnel. The objective is to detect and report irregularities in the area, like intruders, while acquiring their positions and imagery, and transmitting to control station.

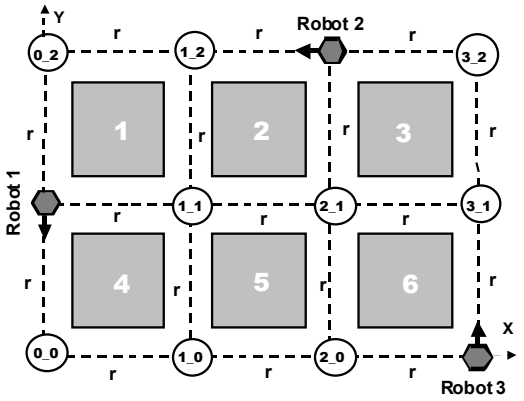


Figure 10: Camp security scenario.

The general picture is shown in Fig. 10, where the camp units (numbered 1 to 6) are simultaneously patrolled by a number of robotic vehicles moving along the paths between and around the buildings.

Distributed Campus Map. The proper routing of vehicles and resolution of possible conflicts between them (like collision avoidance) can be assisted by the creation of a distributed map of the campus area (just reflecting Fig. 10) by the following DSL program (with node names reflecting X-Y coordinates of the crossings, and all links named `r`):

```
create(#3_1; F1=A; r#2_1; F2=A;
 r#1_1; F3=A; r#0_1;(r#0_2; r#1_2;
 r#F3, (r#2_2; r#F2,(r#3_2; r#F1))),
 (r#0_0; r#1_0; r#F3, (r#2_0; r#F2,
 (r#3_0; r#F1))))
```

Random Movement. The next program organizes the duty performance by three parallel processes (which may be executed by three robots) using the created distributed map, with random choice of the next-hop crossing and activation of the external service procedure `move_check_report` to analyze the local security situation while on the move.

```
hop(0_1, 2_2, 3_0); WHERE = CONTENT;
repeat(or((hop(link(random));
 grasp(Mark == nil; Mark = 1);
 (hop(BACK); Mark) = nil;
 move_check_report(CONTENT)), stay))
```

Movement via Predetermined Routes. If to use predetermined routes only, like the ones shown in Fig. 11 (one route using links named `r1` and another one `r2`), the collisions between robots can be avoided in full.

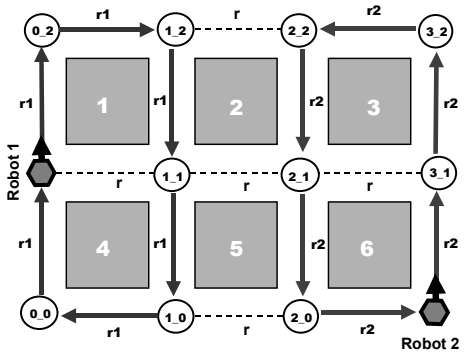


Figure 11: Using predetermined routes.

Additional links `r1` and `r2` in the campus map can be installed by the following DSL program:

```
Linkup((#0_2; r1#1_2; r1#1_1; r1#1_0;
 r1#0_0; r1#0_1; r1#0_2),
 (#3_2; r1#2_2; r1#2_1; r1#2_0;
 r1#3_0; r1#3_1; r1#3_2))
```

And two independent spatial processes navigating the campus via the new links (which may engage two robots) can be organized by the following parallel DSL code:

```
(hop(0_1); Flink = +r1),
(hop(3_0); Flink = +r2);
```

```
WHERE = CONTENT;
repeat(hop(link(Flink));
 move_check_report(CONTENT))
```

Any imaginable combinations of different types of simultaneous movement through the camp (like those by predetermined routes and/or by free, random, wandering) with collision avoidance can also be easily organized in DSL.

5.3 Transport Convoy

Imagine there is a delivery for a camp located in some distance. The objective is to move at least two vehicles to the target location, where only the first one can be manned and the second should follow the route of the first one, on a certain distance from it. We will consider a fully robotic solution for such a convoy, with two and also any number of vehicles, where only the first vehicle knows (and follows) waypoints toward the target location, while others dynamically chaining with, and following the previous ones on the move.

Two-unit Convoy. It is represented by the communicating `Leader` and `Follower`, where the first one defines its movement by a sequence of waypoints, and the second one, regularly requesting the Leader, moves to the positions previously occupied by it, while keeping a certain threshold distance. This is shown in Fig. 12, and by the DSL program that follows.

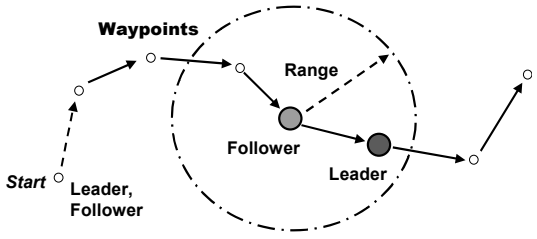


Figure 12: Two-unit convoy.

```
move(start);
(create(Leader);
 Waypoints = (w1, w2, w3, …);
 loop(WHERE = withdraw(Waypoints,1))),
(create(Follower); sling(
 Lcoord = (hop(range, any); WHERE);
 distance(WHERE, Lcoord) > threshold;
 WHERE = Lcoord))
```

Multiple-unit Convoy. A scenario for the convoy with any number of chained processes (to be materialized by robotic units) is described by the following DSL program and depicted in Fig. 13. For this case, only the first process is a pure leader and the last process a pure follower, while all other processes combine both functionalities, i.e. being followers for the previous processes and leaders for the subsequent ones.

```
move(start);
cycle(N < number; create(N += 1));
(NAME == 1; Waypoints = (w1,w2,w3,…);
 Loop(WHERE = withdraw(Waypoints,1))),
(NAME != 1; sling(
 Lcoord = (hop(range,NAME-1); WHERE);
 distance(WHERE, Lcoord) > threshold;
 WHERE = Lcoord))
```

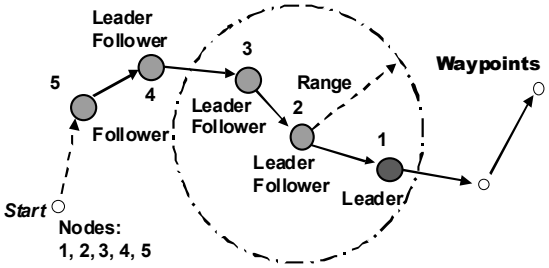


Figure 13: Multiple-unit convoy.

5.4 Transport Mule

Fir this scenario, there are two camps with a certain distance in between, and a cargo with a given weight should be transferred between the camps. We will consider here different possibilities to deliver payload between the camps, using unmanned vehicles as "mules".

In a Single Piece. This may be the case if cargo's weight allows it to be put on a single vehicle, as shown ion Fig. 14.

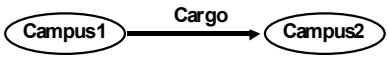


Figure 14: Single piece cargo delivery.

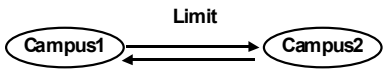


Figure 15: Shuttling delivery.

The related DSL program will be as follows:

```
move(Campus1);
frontal(Cargo) = "substance";
move(Campus2); Store = Cargo
```

Shuttling between Camps. For this option, the process shuttles as often as possible between the two camps after partitioning the cargo into portions for the weight allowed, unless all the cargo is delivered,

as shown in Fig. 15 and by the following program.

```
move(Campus1); frontal(Load);
Cargo = "substance"; Limit = 50;
loop(or((weight(Cargo) > Limit;
  Load = withdraw(Cargo,Limit)),
 (weight(Cargo) > 0; Load = Cargo));
 hop(Campus2); Store += Load;
 hop(Campus1))
```

Multiple, Parallel Delivery. For this case, different processes (vehicles) are considered to be independent from each other, each moving to the destination as quickly as possible on its own (see Fig. 16 and the following program).

```
move(Campus1); frontal(Load);
Cargo = "substance"; Limit = 50;
cycle(or((weight(Cargo) > Limit;
  Load = withdraw(Cargo, Limit)),
 (weight(Cargo) > 0; Load = Cargo)));
move(Campus2); Store += Load
```

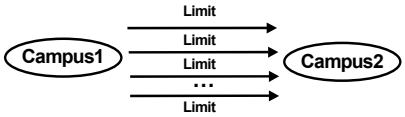


Figure 16: Parallel cargo delivery.

Multiple, Convoy Delivery. For this scenario, the vehicles, each with a limited partition of cargo, are dynamically chaining in a column for a cohesive movement towards the destination (see Fig. 17 and the subsequent DSL program).

```
move(Campus1); frontal(Load);
Cargo = "substance"; Limit = 50;
cycle(or((weight(Cargo) > Limit;
   Load = withdraw(Cargo, Limit)),
  (weight(Cargo) > 0; Load = Cargo));
 create(N += 1));
(NAME == 1; move(Campus2)),
(NAME != 1; loop(WHERE != Campus2;
  WHERE = (hop(NAME-1); WHERE));
Store += Load
```

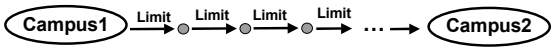


Figure 17: Delivery in a convoy.

5.5 Explosive Ordnance Disposal

Explosive Ordnance Disposal (EOD) means the detection, identification, onsite evaluation, rendering safe, recovery, and final disposal of Unexploded Ordnance (UXO) including detonation and burning. It is often said that the EOD operation is a 3 Ds one, which is Dangerous, Dirty and Demanding (or Difficult) job. Using robotic vehicles, especially multiple ones, is therefore becoming the most promising EOD option.

Various kinds of EOD scenarios for navigation and examination of the target territory may be offered. We will just hint here on the simplest two options, easily expressible in DSL.

Sequential Territory Search. This represents a single-thread process (oriented on a single vehicle), where the whole territory is incrementally scanned unless all being searched, as described by the following program and depicted in Fig. 18.

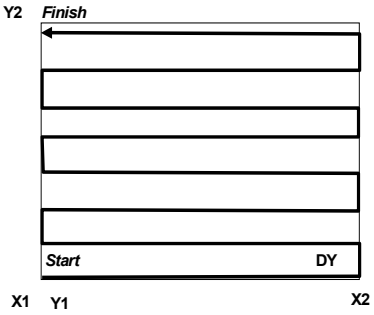


Figure 18: Sequential navigation.

```
X1 =…; X2=…; Y = Y1 =…; Y2 =…; DY =…;
loop(WHERE = (X1,Y); WHERE = (X2,Y);
     (Y += DY) < Y2; WHERE = (X2,Y);
     WHERE = (X1,Y); (Y += DY) < Y2)
```

The sequential coverage of the territory can be organized with minimum waypoints to pass, in a zigzag way, as shown in Fig. 19 and by the following program.

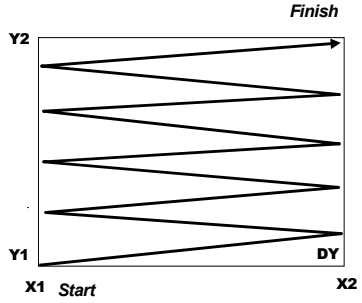


Figure 19: Sequential zigzag navigation.

```
X1 =…; X2=…; Y = Y1 =…; Y2 =…; DY =…;
loop(WHERE = (X1,Y); (Y += DY) < Y2;
     WHERE = (X2,Y); (Y += DY) < Y2)
```

Another solution, starting from the region's periphery and then gradually moving to its center, is shown in Fig. 20, and by the next DSL program.

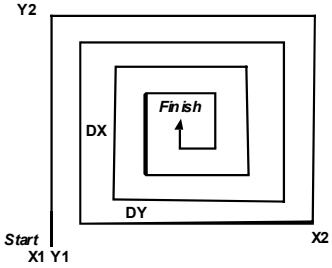


Figure 20: Sequential out-in navigation.

```
X1 = …, X2 = …; Y1 = …; Y2 = …;
DX = …; DY = …; X = X1; Y = Y1;
DDX = X2 – X1; DDY = Y2 – Y1; N = 1;
WHERE = (X,Y);
loop(Y += DDY * N; WHERE = (X,Y);
     X += DDX * N; WHERE = (X,Y);
    (DDX -= DX) > 0; (DDY -= DY) > 0;
     N *= -1)
```

Parallel Territory Search. This can be represented by a number of independent processes, each starting from a different location, and navigating altogether the whole region in parallel, as depicted in Fig. 21, and explained by the DSL program that follows (taking into account that all processes follow *predetermined* routes for this case).

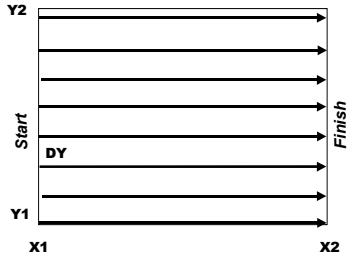


Figure 21: Parallel predetermined navigation.

```
X1 =…; X2=…; Y1 =…; Y2 =…; DY =…;
frontal(Y) = Y1; DDY = 0;
cycle((Y += DDY) < Y2; DDY += DY);
WHERE = (X1,Y); WHERE = (X2,Y)
```

Parallel search of the territory can also be organized in a *random* way, where each process randomly chooses its next hop, also taking into account that the chosen next destination should not have been visited before (at least to look like this, with the help of visual sensors). Parallel random search may have an advantage before predetermined search in that it can eventually cover all the territory despite possible failures of individual processes (robots). Such a search, with processes starting from some initial points (named *c1* to *c5*), where processes also keep certain threshold distance from each other, is shown in Fig. 22 and by the following program.

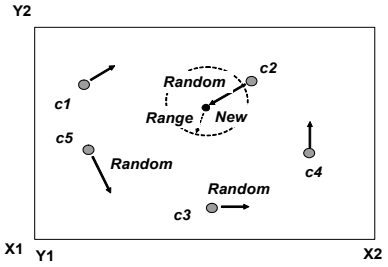


Figure 22: Parallel random navigation.

```
move(c1, c2, c3, c4, c5); Range = …;
X1 =…; X2=…; Y1 =…; Y2 =…; D =…;
loop(New = WHERE + 2(random(-D, D));
 inside(New, (X1, X2, Y1, Y2));
 hop(New, Range) & seen(New) == nil;
 shift_check_act(New))
```

6 OTHER APPLICATIONS

Many other applications of the paradigm are possible, as follows, some of which already investigated, tested, and published (Sapaty, 1999, 2002, 2005, 2007, 2008, 2008a; Sapaty, Sugisaka, Finkelstein, Delgado-Frias, Mirenkov, 2006; Sapaty, Morozov, Sugisaka, 2007).

Emergency Management. Using interpreters installed in massively wearable devices may allow us to assemble workable systems from any wreckage after the disasters, using any remaining communication channels, manual including. These emergent systems can provide distributed self-awareness, collect statistics of casualties, guide the delivery of relief goods, and coordinate collective escape from the disaster zone.

Directed Energy Systems. The technology can provide high flexibility in organizing directed energy (DE) systems, especially in crisis situations, making automatic distributed decisions with the "speed of light" too. It may also help automate the global power dominance by optimized delivery of directed energy into any world points via dynamically organized networks of relay mirrors.

Distributed Avionics. Implanting interpreter copies into main control points of the aircraft may provide a higher, intelligent, layer of its self-analysis and recovery, by the spreading recursive scenarios starting from any point and collecting & fusing key data from other points. The embedded interpretation network with local, dynamic, and emergent links will be fully functional under any damages, especially with wireless communications between

the interpreters. This may always provide global control integrity, even in a physically disintegrating object, helping to save lives and complete missions.

Sensor Networks. Wireless sensors may be dropped from the air massively, as "smart dust". Having a limited communication range, they must operate in a network to do nonlocal jobs in a distributed environment. With the technology offered, we can convert their emergent networks into a universal parallel computer operating in DSL. It can effectively solve complex distributed problems--from just collecting and fusing scattered data to outlining and assembling images of the distributed phenomena like, for example, flooding, smog, flocks of birds, movement of troops, etc.

Advanced Command and Control. In DSL, it is possible to define high-level scenarios concentrating on mission goals and top decision-making while delegating C2 routines, appearing at runtime as a derivative of the mission and environment states, to automatic interpretation. It is also convenient to express in DSL any theoretical and practical issues of advanced C2 explicitly.

Infrastructure Protection. Navigating the systems at runtime, the technology can analyze safety and integrity of critical infrastructures and key resources, establishing protective networked mechanisms throughout them. Other systems can be involved from the WPT layer for emergent infrastructure protection and recovery, including air and space defense, police and army. In relation to energy infrastructures, the technology can help observe power networks from the air or ground, trace electric, gas, or oil supply lines, sensing their states (and, if needed, directly accessing the disaster zones), also providing regular or emergent sentry duties at power installations, etc.

Global and Battlespace Dominance. The DSL scenarios, using any electronic media, can self-spread, outline, and grasp distributed systems of different natures while establishing global dominance over them. They can analyze their internal infrastructures, finding strong and weak points, orient behavior, or destroy the infrastructures or the system as a whole if required. The approach, as an intelligent self-recovering super-virus, which is difficult to discover and kill by traditional means, can effectively employ advanced robotic facilities, like swarms of aerial and ground vehicles, to attack adversarial systems.

7 GESTALT-RELATED

Our approach may be considered as one of the first attempts to formalize and implement the notion of *gestalt* (Wertheimer, 1924), under which the whole dominates over parts (being greater than the sum of them), with parts having sense only in the context of the whole, rather than vice versa. Gestalt theory represented the main departure from atomistic vision of systems at the beginning of the last century. Many existing systems, especially distributed ones, are still based on the concept of predetermined parts (agents) that communicate with each other in an attempt to get the global behavior needed. The latter, with rapidly growing number of agents in complex systems and starting from the agents level is becoming more and more problematic.

Within the approach offered, we have come to quite a different and higher level model of the system organization. Abstracting from system parts and their interactions, which may be emergent and varying at runtime, we can describe the needed global system behavior on a semantic level, where parts and their interactions may not be known in advance, and may dynamically appear (disappear too) just to maintain the global behavior needed.

The technology developed allows us to automatically interpret global system scenarios in any networked systems (comprising internet hosts, laptops, mobile robots, mobile phones, smart sensors, and/or humans themselves). It allows us to get even higher—to describe *what the system should do* on the highest level, where its local and global behavior is a *derivative of this description*, which, in its turn, makes the system structures and operation as a *further derivative*.

8 CONCLUSIONS

A novel ideology and technology, converting any distributed system into a universal spatial machine capable of solving complex problems on itself and on the surrounding environments, has been presented. This conversion can be achieved by implanting into the system sensitive points and its active doers, humans and robots including, of the same copy of a universal control module, communicating with other such modules via available channels. Their entire network, which may be dynamic and emergent, collectively interprets mission scenarios written in a special high-level language, which are defining system's internal and external behavior.

Created and modified on the fly, the scenarios can start from any component, covering the system at runtime through the cooperating interpreters. During the scenario evolution, any operations can be carried out throughout the distributed world, along with the needed movement of code, equipment and artificial or biological doers, humans including, as well as creation and maintenance of physical and virtual infrastructures supporting the missions.

The approach offered can dramatically simplify application programming in distributed systems, especially robotized ones. As can be seen from the examples throughout this paper, programming multi-robot scenarios in distributed and dynamic environments in DSL may not be more difficult than, say, programming of routine data processing tasks in traditional languages like Fortran, C, or Java.

The distributed robotized systems are of rapidly growing importance in many areas, and especially in defense, where robotic swarming on asymmetric battlefields is becoming a major dimension of the new military doctrine for the 21st century (Singer, 2009). The written above is much in line with these trends, allowing us to flexibly combine loose swarming with more classical command and control, which can help gradually transform fully manned into mixed and ultimately totally unmanned systems.

Other prospective applications of this work can be linked with economy, ecology and weather prediction—by using the whole networked world as a spatial supercomputer, self-optimizing its performance.

The approach offered may also be compared with the invention of the first world computers and first high-level programming languages (Zuse, 1948/49; Rojas, 1997). In our case, this computer may not only operate with data stored in a localized memory, but can cover, grasp, and manage any distributed systems, the whole world including, and can work not only with information but with physical matter or physical objects too.

ACKNOWLEDGEMENTS

This work has been funded by the Alexander von Humboldt (AvH) Foundation in Germany. Special thanks to Klaus-Dieter Kuhnert and Matthias Langer for sheltering this activity at the University of Siegen. Years of cooperation with Robert Finkelstein (Robotic Technology Inc., USA) and Masanori Sugisaka (Nippon Bunri University, Japan) contributed much to the ideas expressed in this paper. The support of this ideology and technology by Joaquim Filipe (Escola Superior de Tecnologia, Portugal) and by ICINCO conferences was really invaluable. Encouragement from Stephen Lambacher (Aoyama Gakuin University, Japan) has been appreciated too.

REFERENCES

Kuhnert, K.-D., Krödel, M., 2005. Autonomous Vehicle Steering Based on Evaluative Feedback by Reinforcement Learning. *MLDM*.

M-ELROB, 2008. *Military European Land-Robot Trial*. Hammelburg, Germany.

Rojas, R., 1997. Konrad Zuse's Legacy: The Architecture of the Z1 and Z3. *IEEE Annals of the History of Computing*. Vol. 19, No. 2.

Sapaty, P. S., 1999. *Mobile Processing in Distributed and Open Environments*, John Wiley & Sons. New York.

Sapaty, P. S., 2002. Over-Operability in Distributed Simulation and Control. *The MSIAC's M&S Journal Online*. Winter Issue, Volume 4, No. 2, VA, USA.

Sapaty, P. S., 2005. *Ruling Distributed Dynamic Worlds*, John Wiley & Sons. New York.

Sapaty, P., Sugisaka, M., Finkelstein, R., Delgado-Frias, J., Mirenkov, N., 2006. Advanced IT Support of Crisis Relief Missions. *Journal of Emergency Management*, Vol. 4, No. 4.

Sapaty, P., Morozov, A., Sugisaka, M., 2007. DEW in a Network Enabled Environment. *Proc. International conference Directed Energy Weapons 2007*. Le Meridien Piccadilly, London, UK.

Sapaty, P., 2007. Intelligent management of distributed sensor networks, In *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense VI*, ed. by E. M. Carapezza. *Proc. of SPIE* Vol. 6538, 653812.

Sapaty, P., 2008. Distributed Technology for Global Dominance. *Proc. of SPIE*, *Volume 6981, Defense Transformation and Net-Centric Systems 2008*. Raja Suresh, Ed., 69810T.

Sapaty, P., 2008a. Grasping the Whole by Spatial Intelligence: A Higher Level for Distributed Avionics. *Proc. International Conference Military Avionics 2008*. Cafe Royal, London, UK.

Sapaty, P., 2009. Gestalt-Based Ideology and Technology for Spatial Control of Distributed Dynamic Systems. *Proc. International Gestalt Theory Congress, 16th Scientific Convention of the GTA*. University of Osnabrück, Germany.

Singer, P. W., 2009. *Wired for War: The Robotics Revolution and Conflict in the 21st Century*, Penguin.

Wertheimer, M., 1924. *Gestalt Theory*, Erlangen. Berlin, 1925.

Zuse, K., 1948/49. "Uber den Plankalk, als Mittel zur Formulierung schematisch kombinativer Aufgaben", In *Archiv Mathematik, Band I.*

ITERATIVE FEEDBACK TUNING APPROACH TO A CLASS OF STATE FEEDBACK-CONTROLLED SERVO SYSTEMS

Mircea-Bogdan Rădac, Radu-Emil Precup
Dept. of Automation and Appl. Inf., "Politehnica" University of Timisoara
Bd. V. Parvan 2, 300223 Timisoara, Romania
mircea.radac@aut.upt.ro, radu.precup@aut.upt.ro

Emil M. Petriu
School of Information Technology and Eng., University of Ottawa
800 King Edward, Ottawa, ON, K1N 6N5 Canada
petriu@site.uottawa.ca

Stefan Preitl, Claudia-Adina Dragoş
Dept. of Automation and Appl. Inf., "Politehnica" University of Timisoara
Bd. V. Parvan 2, 300223 Timisoara, Romania
stefan.preitl@aut.upt.ro, claudia.dragos@aut.upt.ro

Keywords: Iterative Feedback Tuning, Servo systems, State feedback control systems.

Abstract: An original control structure dedicated to a class of second-order state feedback control systems is presented in the paper. The controlled processes are accepted to be characterized by second-order servo systems with integral component. Optimal state feedback control systems are designed for those processes making use of the Iterative Feedback Tuning (IFT) approach. The state feedback control system structure is extended with an integral component to ensure the rejection of constant disturbances. A case study concerning the position control of a DC servo system with backlash is included. Real-time experimental results validate the theoretical part of the IFT approach.

1 INTRODUCTION

The second-order servo systems with integral component are applied widely as controlled processes in real-world applications including mechatronics, electrical drives, sub-systems in power plant control systems, positioning systems in manipulators, mobile robots, machine tools, flight guidance and control (Škrjanc et al., 2005; Gomes et al., 2007; Petres et al., 2007; Barut et al., 2008; Costas-Perez et al., 2008; Denève et al., 2008; De Santis et al., 2008; Orlowska-Kowalska and Szabat, 2008; Precup et al., 2008b; Vaščák, 2008). Those controlled processes are acknowledged as particular cases of benchmark systems (Åström and Hägglund, 2000; Isermann, 2003; Horváth and Rudas, 2004; Kovács, 2006). Accepting that they are linearized versions of nonlinear servo systems, the parameters are variable with respect to the operating points. Hence the parameter variation makes their control a challenging task when very good control system performance indices are required. Their control problems become even more challenging when low-cost automation solutions are needed in the design and implementation of the control system structures.

One control solution to cope with the accepted class of processes described is represented by state feedback control systems. Since the main control aims, high performance indices in reference input tracking and regulation with respect to several types of load disturbance inputs, are difficult to be fulfilled, one typical approach is to design optimal control systems. The improvement of the control system performance indices (fore example settling time and overshoot) is enabled by the minimization of appropriately defined objective functions resulting in optimal state feedback control systems. An alternative to the minimization of the objective functions is represented by Iterative Feedback Tuning (IFT) (Hjalmarsson et al., 1994, 1998). IFT

algorithms make use of the input-output data measured from the closed-loop system during its operation to calculate the estimates of the gradients and Hessians of the objective functions. Several experiments are done per iteration and the updated controller parameters are calculated based on the input-output data and the estimates.

The application of IFT to one-degree-of-freedom controllers needs two experiments per iteration. The first experiment is referred to as the normal one and it corresponds to the usual operation of the control system. The second experiment is the gradient one. The reference input in the gradient experiment is the control error of the first experiment. An additional normal experiment is needed in case of two-degree-of-freedom controllers. Even more experiments are needed to tune the state feedback controllers and the Multi Input-Multi Output (MIMO) ones. So it is natural to strive for the alleviation of the number of experiments (Hjalmarsson and Birkeland, 1998; Hjalmarsson, 1999; Jansson and Hjalmarsson, 2004).

The paper aims three main contributions. The first contribution of the paper is the proposal of an IFT algorithm resulting in a method to obtain the partial derivatives needed in the calculation of the gradient of the objective function in state feedback control systems. The second contribution concerns the new experiments to be done in the IFT of the accepted class of second-order state feedback control systems dedicated to servo systems. The third contribution involves the highlighting of the specific aspects related to the actuator saturation problem proved by the low-cost implementation and the real-time experimental results included. The main advantages of the contributions are the simplification of the experiments and the smooth decrease of the objective function. Thus the local minimum will be reached.

The paper treats the following topics. The controlled processes and the new IFT algorithm dedicated to the accepted class of state feedback control system are presented in Section 2. Next, Section 3 points out original and attractive aspects concerning the actuator saturation problem. A case study concentrated on the state feedback position control of a DC servo system with backlash is described in Section 4. The real-time experimental results validate the IFT algorithm. The conclusions are drawn in Section 5.

2 CONTROLLED PROCESS AND IFT ALGORITHM

The controlled process as part of servo systems is characterized by the following state-space model:

$$\begin{bmatrix}\dot{\alpha}\\ \dot{\omega}\end{bmatrix}=\begin{bmatrix}0 & 1\\ 0 & -\frac{1}{T_s}\end{bmatrix}\begin{bmatrix}\alpha\\ \omega\end{bmatrix}+\begin{bmatrix}0\\ \frac{K_s}{T_s}\end{bmatrix}u,\qquad \begin{bmatrix}y_1\\ y_2\end{bmatrix}=I_2\begin{bmatrix}\alpha\\ \omega\end{bmatrix} \tag{1}$$

where $\alpha=x_1$ is the first state variable usually representing the (angular) position, $\omega=x_2$ is the second state variable usually representing the (angular) speed, u is the control signal, y_1 and y_2 are the controlled outputs, and $\mathbf{I}_2$ is the identity matrix. The two parameters in (1) are $K_S>0$ which is the process gain, and $T_S>0$ which stands for the small time constant or the sum of parasitic time constants.

The two transfer functions from u to ω and u to α are $P_{\omega,u}(s)$ and $P_{\alpha,u}(s)$, respectively:

$$P_{\omega,u}(s)=\frac{K_s}{(1+sT_s)},\ P_{\alpha,u}(s)=\frac{K_s}{s(1+sT_s)}. \tag{2}$$

Therefore the integral component can be observed in (2) when $\alpha=x_1$ is taken as controlled output. Such situations correspond to positioning systems.

The state feedback control system structure is presented in Figure 1. The dotted connection highlighted is valid only when the experiments specific to IFT are done. That connection is not applied during the normal system operation.

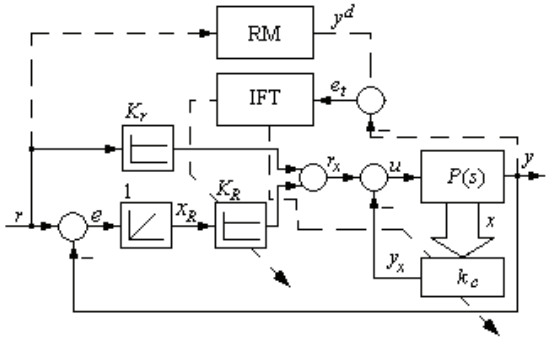


Figure 1: IFT-based state feedback control system structure.

The main variables and blocks illustrated in Figure 1 represent: IFT – the IFT algorithm, RM – the reference model, $\mathbf{x}=[x_1 \quad x_2]^T\in R^2$ – the state vector (T highlights the matrix transposition),

$\mathbf{k}_c = [K_1 \quad K_2]$ – the state feedback gain matrix, $P(s) = P_{\alpha,u}(s)$ – the transfer function of the controlled process when the controlled output is $y=x_1$, r – the reference input, $e=r-y$ – the control error. The other variables will be presented in the sequel.

If the state feedback gain matrix is regarded as a controller, then use will be made of its parameters to minimize the tracking error e_t between the system output y and the reference model output y^d. Let J be a simple objective function defined over a finite time horizon N:

$$J(\boldsymbol{\rho}) = \frac{1}{2N}\sum_{t=1}^{N}(e_t(\boldsymbol{\rho}))^2 , \tag{3}$$

where $\boldsymbol{\rho} \in R^m$ is the parameters vector containing at least the parameters of $\mathbf{k}_c$ and e_t is the tracking error:

$$e_t(\boldsymbol{\rho}) = y(\boldsymbol{\rho}) - y^d . \tag{4}$$

The IFT results (Hjalmarsson et al., 1994, 1998; Pfeiffer et al., 2006) are employed to find the solution ρ^* to the optimization problem

$$\boldsymbol{\rho}^* = \arg\min_{\rho \in SD} J(\boldsymbol{\rho}) , \tag{5}$$

where several constraints can be imposed regarding the process and the closed-loop system. One constraint concerns the stability of the system and *SD* represents the stability domain (Precup et al., 2008).

Solving the optimization problem (5) requires finding the parameters vectors that make the gradient equal to zero:

$$\frac{\partial J}{\partial \boldsymbol{\rho}} = [\frac{\partial J}{\partial \rho_1} \quad \dots \quad \frac{\partial J}{\partial \rho_m}]^T = 0 . \tag{6}$$

Making use of (3) and (4) the equation (6) will be transformed into

$$\frac{1}{N}\sum_{t=1}^{N}\frac{\partial y}{\partial \boldsymbol{\rho}}^T [y(\boldsymbol{\rho}) - y^d] = 0 . \tag{7}$$

The partial derivatives $\frac{\partial y}{\partial \rho_i}$ should be calculated to obtain the components of the gradient, $\frac{\partial J}{\partial \rho_i}$, $i = \overline{1,m}$. The new IFT approach to be described as follows will employ specific experiments to obtain those components. Use will be made of the following notation:

$$\alpha' = \frac{\partial \alpha}{\partial \rho_i} \tag{8}$$

to highlight the partial derivative of the variable α taken with respect to ρ_i and obtain the simplicity of the presentation.

The state-space model (1) can be reconsidered by including one additional state variable to the state variable. That variable is $x_3=x_R$ and it corresponds to the integrator inserted into the control system structure. Thus its gain K_R will be subject to IFT as it is shown in Figure 1. The extended state-space model of the process is

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\omega} \\ \dot{x}_R \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -K_s K_1 & -\frac{1}{T_s} - K_s K_2 & K_s K_R \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \omega \\ x_R \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_s}{T_s} K_r \\ 1 \end{bmatrix} r , \quad \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = I_3 \begin{bmatrix} \alpha \\ \omega \\ x_R \end{bmatrix} \tag{9}$$

where the parameter K_r is not included in the tuning scheme. Its value is set prior to the application of IFT. One way to choose K_r is to keep a connection between the steady-state value of r and the steady-state value of r_x for which the desired r can be tracked by the steady-state value of y. That value of r_x can be subject to the experimental identification of the state feedback control system.

The preparation of the experimental scheme needed in the calculation of the gradient starts with the reconsideration of the input-output relations specific to the control system structure presented in Figure 1. Observing that generally

$$\begin{aligned} &\mathbf{y} = \mathbf{P}u \\ &u = r_x - \mathbf{k}_c\mathbf{x} = r_x - \mathbf{k}_c\mathbf{y} \text{ for } \mathbf{y} = \mathbf{I}_3\mathbf{x} \end{aligned} , \tag{10}$$

the following relationships hold:

$$\begin{aligned} &r_x = K_r r + K_R x_R , \; u = K_r r + K_R x_R - \\ &- K_1 x_1 - K_2 x_2 = K_r r + \mathbf{K}_c \mathbf{x}_E , \\ &\mathbf{K}_c = [-K_1 \quad -K_2 \quad K_R], \mathbf{x}_E = [x^T \quad x_R]^T \end{aligned} . \tag{11}$$

Next the gradient of y with respect to each parameter can be calculated, where the parameters are the m=3 components of the parameters vector

$$\boldsymbol{\rho} = [K_1 \quad K_2 \quad K_R]^T . \tag{12}$$

Since y and u are functions of ρ it is justified to apply

$$\mathbf{y}' = \mathbf{P}u', \tag{13}$$

leading to

$$u' = \mathbf{K}_c'\mathbf{x}_E + \mathbf{K}_c\mathbf{x}_E'. \tag{14}$$

In addition, accepting the MIMO formalism suggested in (10), the following relationship can be expressed:

$$u' = \mathbf{K}_c'\mathbf{y} + \mathbf{K}_c\mathbf{y}'. \tag{15}$$

Equation (15) is of great importance for the new approach. The first term in the right-hand side of (15), $\mathbf{K}_c'\mathbf{y}$, needs to be added to the control signal to obtain the desired experimental scheme. That term contains the unmodified output vector (in the MIMO framework) so the idea is to obtain it from one first initial experiment (Hjalmarsson et. al., 1998). The second term in the right-hand side, $\mathbf{K}_c\mathbf{y}'$, is measured from the control system structure. Therefore the experimental scheme to calculate the gradients results in terms of Figure 2 (without the blocks RM and IFT for the sake of simplicity).

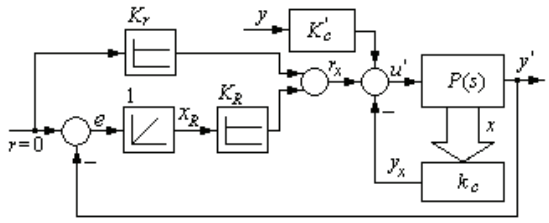


Figure 2: Experimental scheme to calculate the gradients in the IFT-based state feedback control system structure.

The block $\mathbf{K}_c'$ in Figure 2 plays the role of filter. It differs from one experiment to another one depending on the actual parameter with respect to which the gradient is computed.

Since the calculation of the gradients has been derived in the MIMO framework, m+1=4 experiments are done with it. The first experiment, referred to also as the normal one, is done with the control system structure presented in Figure 1 in order to measure the controlled output y. The next m=3 experiments, called the gradient experiments, are done with the experimental scheme presented in Figure 2. These experiments are done separately for each parameter in $\mathbf{K}_c$ (defined in (11)) considering the zero values of the other m–1=2 parameters (because their derivatives with respect to the current parameter are zero).

Once the experiments are done the parameters vector must be updated. Newton's algorithm is generally used as one convenient technique which iteratively approaches a zero of a function without knowledge of it's expression. The update law to calculate the next parameters vector $\boldsymbol{\rho}^{i+1}$ is

$$\boldsymbol{\rho}^{i+1} = \boldsymbol{\rho}^{i} - \gamma_i \mathbf{R}_i^{-1} est[\frac{\partial J}{\partial \boldsymbol{\rho}}(\rho^i)], \tag{16}$$

where i is the index of the current iteration / experiment, γ_i is the step size, $est[\frac{\partial J}{\partial \boldsymbol{\rho}}(\rho^i)]$ is the estimate of the gradient, and the regular matrix $\mathbf{R}_i$ can be the estimate of the Hessian matrix (positive definite) or the identity matrix. The identity matrix is employed when simple implementations are needed.

Making use of all aspects presented before the new IFT algorithm consists of the following steps to be performed per iteration:

Step 1. Do the normal experiment and measure $\mathbf{y}$ based on the control system structure presented in Figure 1. Next do the three gradient experiments making use of the experimental scheme presented in Figure 2 and measure the closed-loop system output that gives the gradient of the controlled output, $\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \mathbf{y}'$.

Step 2. Calculate the output of the reference model, $\mathbf{y}^d$, in terms of the control system structure presented in Figure 3.

Step 3. Calculate the estimate of the gradient of the objective function:

$$est[\frac{\partial J}{\partial \boldsymbol{\rho}}(\rho^i)] = \frac{1}{N}\sum_{t=1}^{N} \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}^T [\mathbf{y}(\boldsymbol{\rho}) - \mathbf{y}^d]. \tag{17}$$

Step 4. Calculate the next set of parameters $\boldsymbol{\rho}^{i+1}$ according to the update law (16).

Three aspects can be highlighted with respect to the above presented IFT algorithm. First, prior to the four steps the designer should set the step size, the reference model and the initial controller parameters in the vector ρ^0. Second, the first task of the state feedback controller is to ensure an initially stable control system. The pole placement design can be used with this regard. Third, the estimate of the Hessian matrix should be calculated in the step 3 is it is used as the matrix $\mathbf{R}_i$ in the update law (16) or an additional experiment can be employed with this regard.

3 ACTUATOR SATURATION PROBLEM

In many cases the actuator is characterized by a nonlinear input-output map caused by the actuator saturation. That is a problem because it introduces usually nonlinear behaviours in the evolution of the process. Hence it should be avoided. When making use of the integrator in the controller the actuator saturation problem becomes important since the actuator that enters a deep saturation region requires usually a longer time to re-enter the active region of normal operation.

Analyzing the structure illustrated in Figure 2 and used in the gradient experiments it is clear that when the state vector is injected in the control signal it may cause saturation. Hence the experiment will be prevented from calculating the correct gradients. In the following, an actuator with the active input range varying from −1 to +1 is considered.

One solution to cope with the above mentioned problem is to design the experiment in such a manner that the actuator never enters saturation. For this, the injected quantity must be in the active region of the actuator's input-output static map. The quantity can be scaled to its maximum value from its evolution. That is obtained by dividing every sample to the maximum absolute value from the sample vector. So it is guaranteed that the new quantity to be injected will be within the accepted domain of the actuator input.

It can be shown as follows how the gradient experiments will be influenced. The general case of MIMO IFT will be considered. First, the scaled, added value to the control is defined as

$$z_s(t) = z(t)/M,\ M = \max_{k=1,N} |\, z(t)\,| \cdot \tag{18}$$

Next the gradient of the control signal with respect to the parameters vector, u', can be expressed in (19) accepting a MIMO control loop with the controller transfer function C:

$$u' = C'(r - y) - Cy' = z - Cy' \cdot \tag{19}$$

Equation (19) is divided by M resulting in the following relationship between the scaled values of the gradients, $u_s' = u'/M$ and $y_s' = y'/M$:

$$u_s' = z/M - Cy_s' \cdot \tag{20}$$

Concluding, dividing (13) by (18) the result will be

$$y_s' = Pu_s' \cdot \tag{21}$$

Practically a scaled value of the estimate of the gradient can be obtained making use of the (20) and (21). After the gradient experiments are done the measured values y_s' are multiplied by M. Thus they will give the normal estimate of the gradient to be used in the iterative minimization of the objective function J.

4 CASE STUDY AND REAL-TIME EXPERIMENTS

The validation of the theoretical approaches is done in terms of a case study consisting of a position control, y=α, of a DC servo system with backlash. The experimental setup illustrated in Figure 3 is built starting with the INTECO DC motor laboratory equipment. It makes use of an optical encoder for the angle measurement and a tacho-generator for the measurement of the angular speed. The tacho-generator measurements are very noisy. The speed can also be observed from the angle measurements. The control system performance indices such as settling time and overshoot can be assessed easily.

The process (1) is characterized by the parameters $K_s = 139.88$ and $T_s = 0.9198\,\mathrm{s}$, obtained after experimental identification. The initial parameters vector has been set to $\rho^0 = [0.0132 \quad 0.0126 \quad 0.005]^T$ which has been obtained to stabilize the system.

A constant reference input has been applied, $r = 150\,\mathrm{rad}$. This allows, without any loss of generality, to pre-tune the parameter K_r at the value $K_r = 0.0133$ and drop it of the variables in the optimization problem (5). That value of K_r has been obtained by steady-state calculation as a gain that connects r with α through the steady-state gain of the inner state-feedback loop. The sampling period has been set to 0.01 s. The following reference model has been considered:

$$G_{RM}(s) = 1/(s^2 + 1.5s + 1) \cdot \tag{22}$$

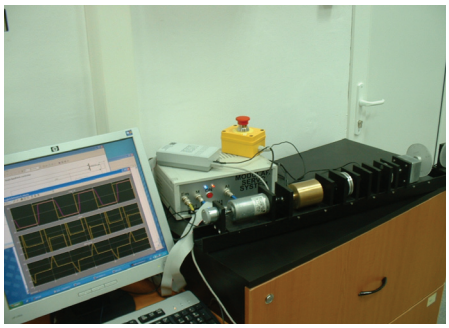


Figure 3: Experimental setup.

Its corresponding pulse transfer function has been obtained for the accepted sampling period. The behaviour of the control system before the application of the IFT algorithm is illustrated in Figure 4.

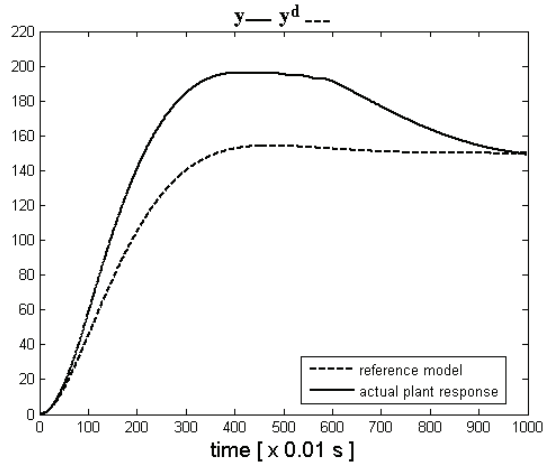


Figure 4: Reference model output and controlled output (position) versus time before IFT.

The IFT algorithm has been applied according to the steps presented in Section 3. The parameters have been set to $\gamma_i = 0.0001$ and $R_i = I_3$. The behaviour of the control system after 12 iterations is presented in Figure 5. The control system performance enhancement is highlighted. It is reflected by smaller overshoot and settling time.

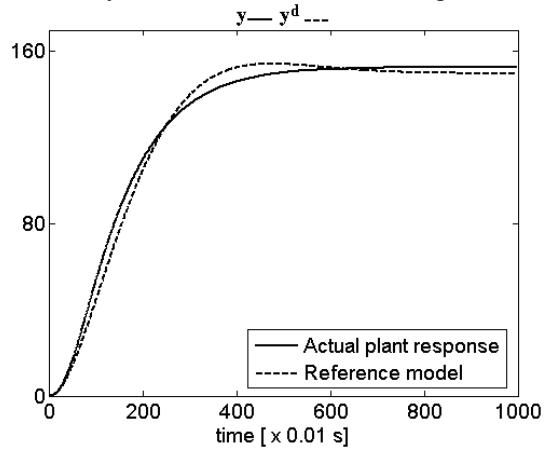


Figure 5: Reference model output and controlled output (position) versus time after IFT.

The variation of the objective function versus the iteration number is illustrated in Figure 6. It shows a good decrease of the objective function and the fact that the number of iterations can be even smaller.

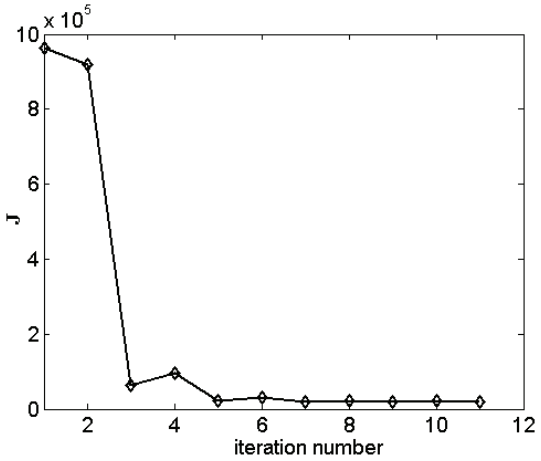


Figure 6: Objective function versus iteration number.

5 CONCLUSIONS

The paper has been presented a new approach to the IFT-based design of state feedback control systems meant for a class of second-order systems with integral component. The new IFT algorithm can be applied without any difficulties to the state feedback control of systems of arbitrary order.

The case study accompanied by real-time experimental results validates the theoretical approaches. The control system designed exhibits better performance indices compared to the situation prior to the application of the IFT algorithm.

The static and kinetic frictions were neglected. They can result in the nonlinearity of the input-output static map $\omega = f(u)$. The idealization considered here simplifies the model to be handled easily because the nonlinearity is not strong.

The first limitation of the proposed IFT approach concerns the tuning of the initial parameters of the controller (grouped in the vector ρ^0). That problem is not simple if nonlinear processes are involved. The second limitation is that the global optimum cannot be guaranteed. Hence only quasi-optimal state feedback control systems can be designed.

The presence of the parameter K_r presented in Figure 1 and Figure 2 is not mandatory because the integrator acts in the direction of error alleviation. So the control system structure can be simplified. However its presence is important because it can influence the initial control error with effects on the convergence of the IFT algorithm.

The future research will be focused on: the consideration of more complex objective functions to include the control signal, the state and output

sensitivity functions as well, the generalization to nonlinear processes (Cottenceau et al., 2001; Johanyák and Kovács, 2007; Savaresi et al., 2006; Andrade-Cetto and Thomas, 2008; Giua and Seatzu, 2008; Precup et al., 2008a; Dolgui et al., 2009) including MIMO servo systems, and the mapping of the results from the linear case onto the parameters of the fuzzy controllers in the framework of state feedback fuzzy control systems. The convergence analysis of all IFT algorithms is needed.

ACKNOWLEDGEMENTS

The paper was supported by the CNMP & CNCSIS of Romania. The first and fifth authors are doctoral students with the "Politehnica" University of Timisoara, Romania, and also SOP HRD stipendiaries co-financed by the European Social Fund through the project ID 6998.

REFERENCES

Andrade-Cetto, J., Thomas, F., 2008. A wire-based active tracker. *IEEE Transactions on Robotics*. 24, 642-651.

Åström, K. J., Hägglund, T., 2000. Benchmark systems for PID control. In *Preprints of IFAC PID'00 Workshop*. Terrassa, Spain, 181-182.

Barut, M., Bogosyan, S., Gokasan, M., 2008. Experimental evaluation of braided EKF for sensorless control of induction motors. *IEEE Transactions on Industrial Electronics*. 55, 620-632.

Costas-Perez, L., Lago, D., Farina, J., Rodriguez-Andina, J. J., 2008. Optimization of an industrial sensor and data acquisition laboratory through time sharing and remote access. *IEEE Transactions on Industrial Electronics*. 55, 2397-2404.

Cottenceau, B., Hardouin, L., Boimond, J.-L., Ferrier, J.-L., 2001. Model reference control for timed event graphs in dioids. *Automatica*. 37, 1451-1458.

Denève, A., Moughamir, S., Afilal, L., Zaytoon, J., 2008. Control system design of a 3-DOF upper limbs rehabilitation robot. *Computer Methods and Programs in Biomedicine*. 89, 202-214.

De Santis, A., Siciliano, B., Villani, L., 2008. A unified fuzzy logic approach to trajectory planning and inverse kinematics for a fire fighting robot operating in tunnels. *Intelligent Service Robotics*. 1, 41-49.

Dolgui A., Guschinsky, N., Levin, G., 2009. Graph approach for optimal design of transfer machine with rotary table. *International Journal of Production Research*. 47, 321-341.

Giua, A., Seatzu, C., 2008. Modeling and supervisory control of railway networks using Petri nets. *IEEE Transactions on Automation Science and Engineering*. 6, 431-445.

Gomes, L., Costa, A., Barros, J. P., Lima, P., 2007. From Petri net models to VHDL implementation of digital controllers. In *Proceedings of 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON 2007)*. Taipei, Taiwan, 94-99.

Hjalmarsson, H., 1999. Efficient tuning of linear multivariable controllers using Iterative Feedback Tuning. *International Journal of Adaptive Control and Signal Processing*. 13, 553-572.

Hjalmarsson, H., Birkeland, T., 1998. Iterative Feedback Tuning of linear time-invariant MIMO systems. In *Proceedings of 37th IEEE Conference on Decision and Control*. Tampa, FL, 3893-3898.

Hjalmarsson, H., Gevers, M., Gunnarsson, S., Lequin, O., 1998. Iterative Feedback Tuning: theory and applications. *IEEE Control Systems Magazine*. 18, 26-41.

Hjalmarsson, H., Gunnarsson, S., Gevers, M., 1994. A convergent iterative restricted complexity control design scheme. In *Proceedings of 33rd IEEE Conference on Decision and Control*. Lake Buena Vista, FL, 1735-1740.

Horváth, L., Rudas, I. J., 2004. *Modeling and Problem Solving Methods for Engineers*. Burlington, MA: Academic Press, Elsevier.

Isermann, R., 2003. *Mechatronic Systems: Fundamentals*. Berlin, Heidelberg, New York: Springer-Verlag.

Jansson, H., Hjalmarsson, H., 2004. Gradient approximations in Iterative Feedback Tuning for multivariable processes. *International Journal of Adaptive Control and Signal Processing*. 18, 665-681.

Johanyák, Z. C., Kovács, S., 2007. Sparse fuzzy system generation by rule base extension. In *Proceedings of 11th International Conference on Intelligent Engineering Systems (INES 2007)*. Budapest, Hungary, 99-104.

Kovács, G. L., 2006. Management and production control issues of distributed enterprises. In *Proceedings of PROLAMAT 2006 IFIP TC5 International Conference*. Shanghai, China, 11-20.

Orlowska-Kowalska, T., Szabat, K., 2008. Damping of torsional vibrations in two-mass system using adaptive sliding neuro-fuzzy approach. *IEEE Transactions on Industrial Informatics*. 4, 47-57.

Petres, Z., Baranyi, P., Korondi, P., Hashimoto, H., 2007. Trajectory tracking by TP model transformation: case study of a benchmark problem. *IEEE Transactions on Industrial Electronics*. 54, 1654-1663.

Pfeiffer, D., Stephens, R. I., Vinsonneau, B., Burnham, K. J., 2006. Iterative Feedback Tuning applied to a ship positioning controller. In *Proceedings of 18th International Conference on Systems Engineering (ICSE 2006)*. Coventry, UK, 353-358.

Precup, R.-E., Preitl, S., Fodor, J., Ursache, I.-B., Clep, P. A., Kilyeni, S., 2008a. Experimental validation of Iterative Feedback Tuning solutions for inverted pendulum crane mode control. In *Proceedings of 2008 Conference on Human System Interaction (HSI 2008)*. Krakow, Poland, 536-541.

Precup, R.-E., Preitl, S., Rudas, I. J., Tomescu, M. L., Tar,

J. K., 2008b. Design and experiments for a class of fuzzy controlled servo systems. *IEEE/ASME Transactions on Mechatronics*. 13, 22-35.

Savaresi S. M., Tanelli, M., Taroni, F., Previdi, F. Bittanti, S., Prandoni, V., 2006. Analysis and design of an automatic motion-inverter. *IEEE/ASME Transactions on Mechatronics*. 11, 346-357.

Škrjanc, I., Blažič, S., Agamennoni, O., 2005. Interval fuzzy model identification using l_∞-norm. *IEEE Transactions on Fuzzy Systems*, 13, 561-568.

Vaščák, J., 2008. Fuzzy cognitive maps in path planning. *Acta Technica Jaurinensis, Series Intelligentia Computatorica*. 1, 467-479.

A HYBRID SET-UP OPTIMIZATION MODEL FOR TANDEM COLD ROLLING MILL

Mohammad Hadi Mirmohammadi, Hossein Haddad
Department of CRM Automation Engineering, Mobarakeh Steel Company, Isfahan, Iran
mirmohamadi@gmail.com, hossein.haddad@gmail.com

Seyed Mehdi Naghavi
Department of Tandem Cold Rolling Mill, Mobarakeh Steel Company, Isfahan, Iran
shv@mobarakeh-steel.ir

Keywords: Hybrid optimization, Real-coded genetic algorithm, Nelder and Mead simplex method, Set-up optimization model, Tandem cold rolling.

Abstract: Set-up optimization of the rolling process involves several parameters that may lead to complex multi objective optimization problem. Also, it is well known that experience is playing a vital role in the selection of operating parameters in rolling mill. This paper presents a combination of two optimization procedure for a multi objective optimization problem. The first optimization phase is based on Nelder and Mead simplex method which focus on balance of power, force and reduction distribution in set-up planning. Then, a real-coded genetic algorithm based optimization procedure applies on the system as an outer loop to optimize energy consumption and productivity. An experimental result of application to five stand tandem cold rolling mill is presented.

1 INTRODUCTION

Process optimization is the discipline of adjusting a process so as to optimize some specified set of parameters without violating some constraint. The most common goals are minimizing cost, maximizing throughput, and/or efficiency. This is one of the major goals in industrial automation systems.

The optimization of manufacturing and product quality is possible when the effect of each process stage and its influence on the process parameters are known as a process model and an advanced architecture of optimization mechanism apply on this model to achieve optimized parameters.

Recently, a lot of attention has been devoted toward advanced techniques of computational intelligence for Process optimization. However it is well known that experience is playing a vital role in modelling and optimization of complex industrial processes (Venkata & Suryanarayana, 2001).

The tandem cold rolling of metal strip is a complex nonlinear multivariable process whose optimization presents significant challenges to the control design.

Set-up optimization and Scheduling for tandem cold mills has been frequently investigated in the last few years, motivated by the benefits they can provide in terms of quality and productivity improvements (Pires, et. al., 2006). Reductions, speeds, tensions and forces, which must be followed by the control loops, form the main part of the mill set-up (Bryant, 1973). Reduction and tension distribution is the major point for set-up calculation that usually obtains by look-up table or some simple formula based on experience. Therefore by this distribution, optimal set-up may not be achieved. Optimal schedules should result in maximized throughput and minimized operating cost (Wang, et al., 2000).

In the present work, recent developments will be discussed concerning to set-up optimization applied to a continues five stand tandem cold mill at Mobarakeh Steel plant in Iran, which, due to more high quality and productivity market demand, was totally revamped in 2004. The proposed algorithm is composed of two step optimization architecture. The first part is an inner loop which calculates stand reductions and inter-stand tensions based on reduction balance, power balance and force balance.

For the optimization of the set-up, Nelder and Mead simplex algorithm has been employed. The major goal of this part is to balance of power, force and reduction according to rolling condition and experience. The second part is an outer loop which optimizes rolling productivity by using genetic algorithm. In fact, the outer loop evaluates some coefficient of inner optimization loop to maximize throughput and minimized energy consumption.

The study is organized in the following manner. First, section 2 delivers a brief introduction to the architecture of set-up optimization based on Nelder and Mead simplex algorithm. In section 3, we discuss a productivity optimizer as an outer optimization loop that applies on set-up optimization structure described in section 2. Experimental results are explained in section 4. finally, section 5 presents the main conclusions.

2 PRESET BALANCE MODEL

The structure of the inner loop set-up model is presented in figure 1. The preset balance optimizer receives the power and force of each stand and calculates reductions and tensions. In this work the Bryant model is considered for rolling mill process model (Bryant, 1973).

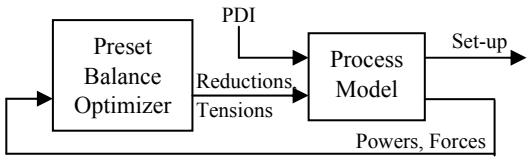


Figure 1: Inner loop set-up model.

The process of set-up calculation consists of power balancing, force balancing and reduction balancing. To balance the power, the required per unit power of each stand should be the same, taken the total available power of the rolling mill as the base power. Thus, the total set-up power should not be greater than the nominal power of the rolling mill. In force balancing the applied forces should be smoothly distributed through the first stands in order to avoid flatness problems. Furthermore, the last stand force is critical due to imposed quality purposes like roughness and flatness. The reduction balancing is based on experience and usually is formed as a look-up table.

The specific tension applied to the strip on each zone should not be greater than one third of the yield strength of the strip on that zone (Bryant, 1973). Furthermore, the distribution of specific tension through the zones should follow the same law as the distribution of reduction through the stands in order to avoid unbalancing of back and front tension on any stand. Also, it is desirable that the stand maximum set-up speed be achieved using the total available power of that stand (Pires, et al., 2006).

To achieve the goals which described we should define a suitable objective function and employ a powerful optimization algorithm. Nelder and Mead simplex method is one of the best algorithm for optimization that is applied in several application related to cold mill set-up optimization.

A detailed description of the simplex method can be found in the work of Pires, et al. (2006). The simplex method, considers the unconstrained minimization of a nonlinear cost function $J = f(x_1, x_2, \ldots, x_n)$ of n variables, without evaluating its derivatives. The minimization step is variable according to the cost function. Briefly speaking, disturbances are introduced in the initial values of x_i and new values of the cost function are calculated, corresponding to each disturbance. Three operations may be accomplished, according to the following steps: reflection, contraction and expansion.

The iterative process is initiated sorting the points x_W, x_N and x_B for which the function has its maximum value J_W, the second maximum value J_N, and the minimum value J_B, respectively.

The average point or centroid x_C is determined finding the average of all points x_i, except x_W. From equation 1 and assuming the minimization step $b = 1$, it results $x = x_R$, known as reflection of x_W with respect to x_C.

$$x = x_C + b(x_C - x_W) \tag{1}$$

The following four cases can then occur:

- If $J_B < J_R < J_N$, then x_W is replaced by x_R and the process is restarted;
- If $J_R < J_B < J_N$, then set $b = 2$ and get $x = x_E$, known as expansion of x_R with respect to x_C. If $J_E < J_B$, x_W is replaced by x_E and a new process is started;
- If $J_N < J_R < J_W$, a contraction is made, generating a vertex $x = x_U$ for which $b = 1/2$. If $J_B < J_U < J_N$, x_W is replaced by x_U and a new process is started;

- If $J_W < J_R$, a contraction with change in direction must be done, generating a vertex $x = x_T$ for which $b = -1/2$. If $J_T < J_W$, x_W is replaced by x_T and a new process is started.

The simplex method requires a starting point not so distant of the optimum point as a condition to converge; the use of empirical laws during initialization or using beta factor algorithm which is described in Pires, et al. (2006) help to reach this objective. From this point, the simplex method calculates the initial cost function and allow for disturbance in reductions and tensions, which result in new values of power, forces and tensions and subsequently in new value for the cost function.

In order to define a suitable objective function for the tandem mill, power, force, reduction and tension were assumed to be the most important variables to form the objective function. Therefore, the objective function was then conceived as:

$$J = k^P \sum_{i=1}^{5} J_i^P + k^F \sum_{i=1}^{5} J_i^F + k^R \sum_{i=1}^{5} J_i^R + \sum_{j=1}^{4} J_j^T \quad (2)$$

Where J_i^P is the power balance cost function, J_i^F is force balance cost function and J_i^R is the reduction balance cost function of stands i = 1, 2, 3, 4, 5, respectively. J_j^T is the tension balance cost function of zones j = 1, 2, 3, 4 between two consecutives stands.

k^P, k^F and k^R present the weight of each cost function of the objective function. We will see later that these coefficients come from the outer optimization loop.

The cost function of power, force, reduction and tension are given by

$$J_i^P = K_i^P \left(\frac{P_i - \frac{P_i^{\max} + P_i^{\min}}{2}}{\frac{P_i^{\max} - P_i^{\min}}{2}} \right)^{N_i^P} \quad (3)$$

$$J_i^F = K_i^F \left(\frac{F_i - \frac{F_i^{\max} + F_i^{\min}}{2}}{\frac{F_i^{\max} - F_i^{\min}}{2}} \right)^{N_i^F} \quad (4)$$

$$J_i^R = K_i^R \left(\frac{R_i - \frac{R_i^{\max} + R_i^{\min}}{2}}{\frac{R_i^{\max} - R_i^{\min}}{2}} \right)^{N_i^T} \quad (5)$$

$$J_i^T = K_i^T \left(\frac{T_i - \frac{T_i^{\max} + T_i^{\min}}{2}}{\frac{T_i^{\max} - T_i^{\min}}{2}} \right)^{N_i^T} \quad (6)$$

N_i^P, N_i^F, N_i^T and K_i^P, K_i^F, K_i^T are exponents and coefficients of cost functions of power, force and tension, respectively. The maximum and minimum limits for power ($P_i^{\max}$ and $P_i^{\min}$), force ($F_i^{\max}$ and $F_i^{\min}$), reduction ($R_i^{\max}$ and $R_i^{\min}$), and tension ($T_i^{\max}$ and $T_i^{\min}$), are normally defined based on the process analyst knowledge. The objective function is strongly penalized if $P < P^{\min}$ or $P > P^{\max}$ for power and also there is the same situation for force, reduction and tension.

The algorithm described above is executed several times until a minimum value for the cost function is reached. The criterion for stopping this process is the number of iterations or a given incremental reduction of cost function between two consecutive iterations. Consequently one set of reduction and tension distribution will be achieved that will be used in other set-up values calculation.

3 HYBRID SET-UP MODEL

In section 2, the optimization was done to balance the main parameters of rolling but it doesn't guarantee to maximize throughput and efficiency. So in this paper, we present an additional optimization loop as an outer loop using genetic algorithm to optimize energy consumption and rolling speed.

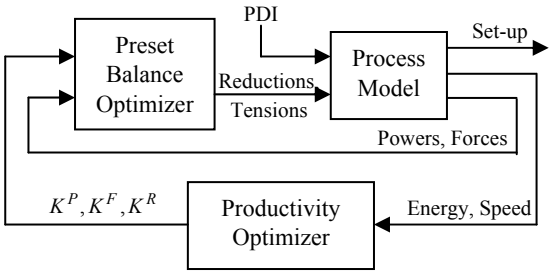


Figure 2: Hybrid set-up model.

Figure 2 shows the structure of hybrid optimization method. The required energy and rolling speed are feedback parameters from process model and is used in productivity optimizer. The outputs of this optimizer are k^P, k^F and k^R which is used as coefficients of objective function in balance optimization. To minimize energy consumption and maximize rolling speed we consider the objective function as:

$$J = \sum \left(\alpha . E + \beta (v_{\max} - v)^2 \right) \quad (7)$$

In this equation, E demonstrate required energy consist of reduction and tension energy and evaluate by rolling energy model. The parameter v is the maximum allowable rolling speed corresponding to reduction distribution and power of motors.

Genetic algorithms (GAs) are gradient free parallel-optimization algorithms that use a performance criterion for evaluation and a population of possible solutions to search for a global optimum. These structured random search techniques are capable of handling complex and irregular solution spaces (Setnes & Roubos, 2000). GAs are inspired by the biological process of Darwinian evolution where selection, mutation, and crossover play a major role. Good solutions are selected and manipulated to achieve new and possibly better solutions. The manipulation is done by the genetic operators that work on the chromosomes in which the parameters of possible solutions are encoded. In each generation of the GA, the new solutions replace the solutions in the population that are selected for deletion.

We consider real-coded GAs. Binary coded or classical GAs are less efficient when applied to multidimensional, high-precision or continuous problems. The bit strings can become very long and the search space blows up. Furthermore, central processing unit (CPU) time is lost to the conversion between the binary and real representation. Other alphabets like the real coding can be favourably applied to variables in the continuous domain. In real-coded GAs, the variables appear directly in the chromosome and are modified by special genetic operators. Various real-coded GAs were recently reviewed in Herrera and Lozano (1998).

The chromosome representation determines the GA structure. We encode the parameters of outer loop in a chromosome as Eq. (8) where $l = 1, \cdots, L$ and L is the size of chromosomes population.

$$S_l = [K_{Pl}, K_{Fl}, K_{Rl}] \quad (8)$$

The selection function is used to create well-performing chromosomes which have a higher chance to survive. The roulette wheel selection method is used to select n_C chromosomes for operation.

Two classical operators, simple arithmetic crossover and uniform mutation and four special real-coded operators are used in the GA. These operators have been successfully applied in the work of Setnes and Roubos (2000) and Michalewicz (1994).

For crossover operations, the chromosomes are selected in pairs. In Simple arithmetic crossover two chromosomes are crossed over at the random position. Whole arithmetic crossover creates a linear combination of two chromosomes as:

$$\begin{aligned} S_v^{t+1} &= r.S_v^t + (1-r).S_W^t \\ S_w^{t+1} &= r.S_w^t + (1-r).S_v^t \end{aligned} \quad (9)$$

In this section, $r \in [0,1]$ and is a random number. Heuristic crossover is another kind of a pair chromosomes combination such that:

$$\begin{aligned} S_v^{t+1} &= S_v^t + r.(S_w^t - S_v^t) \\ S_w^{t+1} &= S_w^t + r.(S_v^t - S_w^t) \end{aligned} \quad (10)$$

For mutation operations, single chromosomes are selected. In Uniform mutation a random selected element is replaced by a random number in the range of element. Multiple uniform mutations is uniform mutation of n randomly selected elements and in Gaussian mutation all elements of a chromosome are mutated such that a random number drawn from a Gaussian distribution with zero mean will be added to each element.

In this paper the chance that a selected chromosome is used in a crossover operation is 95% and the chance for mutation is 5%. When a chromosome is selected for crossover (or mutation) one of the used crossover (or mutation) operators are applied with equal probability. The search space of elements in chromosomes is determined in the range between 0 and 1.

4 EXPERIMENTAL RESULTS

As an experimental work, we implemented the described algorithm on five stand tandem mill of Mobarakeh Steel plant, Iran. In the first step, the Preset Balance Model was implemented to produce optimal set-up values based on objective function presented in Eqs. (2)-(6) by using the values for the

Table 2: Optimization results using Preset Balance Model and Hybrid Set-up Model.

	Preset Balance Model						Hybrid Set-up Model					
	Red. (%)	Tens. (ton)	Speed (mpm)	Force (ton)	Power (kw)	Energy (kwh/t)	Red. (%)	Tens. (ton)	Speed (mpm)	Force (ton)	Power (kw)	Energy (kwh/t)
Zone 0		13	273					13	288			
Stand 1	28.8			865	1724	4.29	32.4			949	2656	6.26
Zone 1		25.9	383					22.7	427			
Stand 2	28.4			890	4560	11.35	26.7			1074	4560	10.74
Zone 2		19.4	535					18.85	582			
Stand 3	26.1			854	4487	11.18	24			846	4300	10.12
Zone 3		16.83	723					16.82	766			
Stand 4	21.1			876	4481	11.15	20.1			795	4375	10.29
Zone 4		16.72	917					16.95	957			
Stand 5	5.36			570	1831	4.58	7			623	2103	4.95
Zone 5		2.86	969					2.85	1029			
Total						42.55						42.36

coefficients and exponents adjusted for each stand that are presented in Table 1. Adjustment of these coefficients has been done by focus on a regular distribution of reductions, power and forces based on rolling situation and experience.

Table 1: Exponents and coefficients of cost functions.

	Std.1	Std.2	Std.3	Std.4	Std.5
K_i^P	1	1	1	1	1
N_i^P	18	18	18	18	10
K_i^F	0.001	0.001	0.001	0.001	1
N_i^F	18	18	18	18	18
K_i^R	1	0.1	0.1	0.1	1
N_i^R	10	10	10	10	10
K_i^T	1	1	1	1	-
N_i^T	5	5	5	5	-

For the second phase, the hybrid set-up model with proposed objective function in Eq. (7) was implemented. We apply a real-coded GA with the population size (L) equal to 20 and $n_C = 5$ for operation in each iteration. The chance of crossover operation is 95% and for mutation is 5%.

The values summarized in Table 2 show the final result of set-up calculation according to the preset balance model in the right side and hybrid set-up model in the left side of table.

By using preset balance model, the balance of power, force and reduction is obtained after approximately 100 iteration but the maximum obtained speed value will be 969 (m/min) while the total required energy is equal to 42.55 (kwh/t).

The implementation of the hybrid set-up model produced, as shown in the right part of Table 2, a significant improvement for the global performance. The rolling speed will increase to 1029 (m/min) and the required energy will decrease to 42.36 (kwh/t). However the balance of power, force and reduction is a little changed but we can obtain more throughputs by using less energy that provide higher productivity.

5 CONCLUSIONS

In this paper we have described a hybrid optimization procedure for set-up generation of tandem cold rolling mill. In order to complexity of rolling process and the role of experience in this process, we should solve a multi objective problem to calculate optimum set-up values. So, we has presented a hybrid algorithm consists of two optimization model.

The preset balance optimization model is based on Nelder and Mead simplex method which optimizes the balance of power, force and reduction of stands. The simplex method, considers the unconstrained minimization of a nonlinear cost function, without evaluating its derivatives. By using empirical laws and beta factor algorithm we can find starting point not so distant of the optimum point which helps us to reach the optimum solution in a few iteration.

The hybrid set-up model appears as an outer loop which minimizes the energy consumption and maximizes rolling speed through evaluation of coefficient of objective function related to preset balance model.

The optimization algorithm used in this model is based on Genetic algorithm and to increase the

efficiency and decrease the processing time, we have proposed a real-coded genetic algorithm which employs some special kind of crossover and mutation operation to reduce the calculation time.

The proposed optimization approach was successfully applied to five stand tandem cold rolling mill, located at mobarakeh steel plant of Iran. Experimental results show that the obtained set-up leads to high quality and productivity in the tandem cold mill.

REFERENCES

Pires, C. T. A., Ferreira, H.C., Sales, R.M., and Silva, M.A., (2006) 'Set-up optimization for tandem cold mill: A case study', *Journal of Materials Processing Technology,* vol. 173, pp.368-375, ELSEVIER Press.

Setnes, M., Roubos, H., (2000) 'GA-Fuzzy Modelling and Classification: Complexity and Performance', *IEEE Transactions on Fuzzy Systems,* vol. 8, no. 5, October, pp. 509-522.

Michalewicz, Z., (1994) *Genetic Algorithms + Data Structures = Evolution Programs*, 2nd edition, New York: Springer-Verlag.

Herrera, F., Lozano, M. and Verdegay, J. L., (1998) 'Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis', *Artificial Intelligence Rev.*, vol. 12, pp. 265–319.

Venkata, R.N., Suryanarayana, G., (2001) 'A set-up model for tandem cold rolling mills', *Journal of Materials Processing Technology*, vol. 116, October, pp. 269-277, ELSEVIER Press.

Wang, D. D., Tieu, A. K., De Boer, F.G., Yuen, W. Y. D., (2000) 'Toward a heuristic optimum design of rolling schedules for tandem cold rolling mills', *Engineering Applications of Artificial Intelligence*, vol. 13, August, pp. 397-406, ELSEVIER Press.

Bryant, G. F., (1973) *Automation of Tandem Mills,* London: Iron and Steel Institute.

CONTACTLESS TORQUE SENSOR

Mechatronic Principle and Prototype Development for Automotive Applications

Manfred Brandl
Austriamicrosystems AG, Schloss Premstätten, Austria
manfred.brandl@austriamicrosystems.com

Franz Haas, Reinhard Marik
FH CAMPUS 02, Department of Automation Technology, Körblergasse 126, Graz, Austria
franz.haas@campus02.at, reinhard.marik@campus02.at

Keywords: Torque sensor, Inductive principle, Electric steering system, Finite element method.

Abstract: In this paper fundamentals and the prototype development of a new contactless torque sensor are presented. The whole device can be divided into a mechanical and an electromagnetic system. The prototype is designed for the torque range of an electric power assisted steering. The basic idea is the transformation of the shaft twist under torque load into a translational movement of the middle part of the sensor sleeve that is measured by an inductive measuring device. An additional aspect is the low cost manufacturing and assembling of the whole system. The interface electronics ensures high linearity and accuracy and is therefore very suitable for this application. This new sensor concept is very robust and self-compensating for all kinds of relative positioning tolerances ranging from temperature change to external forces.

1 INTRODUCTION

The object of this publication is to present a new contactless torque sensor concept based on a highly reliable inductive sensing principle for automotive applications to overcome the drawbacks of present solutions.

Recent automotive fuel saving and power train techniques are requiring highly reliable but low cost torque sensors. One example application is electric power assisted steering system (Graßmann, 2003) to measure input torque. Others are open or closed loop drive train applications such as torque vectoring, power control or power assisted bicycles.

2 TORQUE SENSOR PRINCIPLE

Torque measurement on rotating shafts leads to a specific design requirement. The torque information has to be transmitted in a wireless non-contact manner from the rotating shaft to the static readout environment. The technical solution is to pick up the torque information on the rotating shaft and transmit it to a stationary receiver either through electric, magnetic or electromagnetic field.

Today's sensing techniques are:

- Strain gauge or magneto-elastic sensor on shaft surface to measure strain in 45-degree direction versus shaft axis;
- A compliant torsion bar with optical, resistive, inductive or magnetic (Angleviel, 2005) measurement of the relative angle between both ends. This principle is used in modern electromechanical steering systems (Heißing, 2008) and fulfils the requirements of safety and reliability.

At present torque sensors are mostly used in industrial drive train applications, in robotics and in bench test measurement equipment.

For this new concept the system can be divided into two domains:

- Mechanical domain;
- Electromagnetic domain.

2.1 Mechanical Domain

The twist of a steel shaft under load is transformed to an axial displacement (Jacobsen, 1944). This effect and the core part of the torque sensor are shown in Figure 1. The applied torque causes a shaft

twist that is induced into a specially designed sleeve. This part, for instance made of aluminium, is perfectly attached to the shaft at two cylindrical faces with a constant distance. The connections of the middle part to the outside parts represent thin rods that translate the twist into a translational movement (Δx) along the shaft rotation axis.

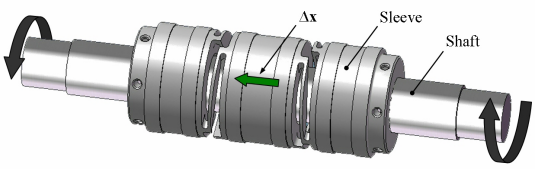


Figure 1: Sleeve with the fixation parts, middle part and connection rods.

Possible manufacturing processes are chipping technologies (turning and milling), powder injection moulding or welded composite from punched parts depending on the production volume.

2.2 Electromagnetic Domain

Two fixed ferromagnetic sleeves, one displaceable centre sleeve and the two axial air gaps as part of the rotor are shown in Figure 2.

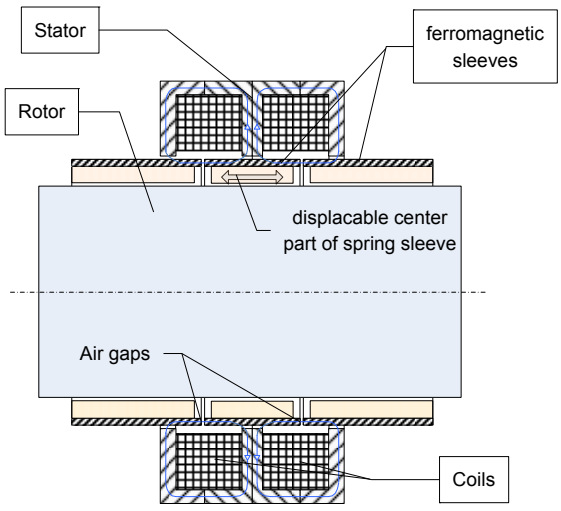


Figure 2: Electro magnetic circuit formed by sensor rotor and sensor stator.

The rotor forms a differential inductive half bridge circuit (see Figure 3) with two fixed coils and its respective ferromagnetic field concentrators as a stator.

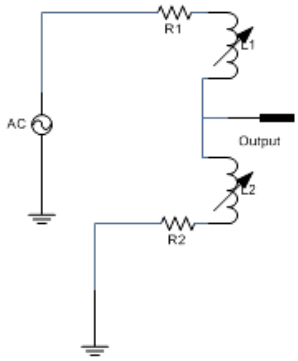


Figure 3: Sensor inductances and its parasitic resistances connected in series to form a half bridge.

The stationary coils and their ferromagnetic housings are placed concentric to the shaft in the middle of the rotor with the two air gaps. When torque is applied, one air gap increases and the other one gets smaller by the same amount. The two coils are connected in series and excited by a common electrical signal. The reluctance ratio determines the output voltage of the inductive half bridge. The magnetic circuit is designed in such a way that a significant portion of the total reluctance is determined by the axial air gaps between the rotor sleeves. Any differential change of air gap due to applied torque leads to a non-balanced bridge and an output signal linear with displacement in case of small deflections.

Because of the symmetric arrangement of the magnetic ring elements all air gap changes due to shaft bending, shaft expansion or shaft compression won't lead to an output signal. Nor will concentricity errors of the rotor influence the output signal. Even axial simultaneous displacement of the entire rotor like a bearing play will not lead to reluctance difference in the first instance. Most important is that also shaft rotation doesn't affect the balance of the electromagnetic circuit, either.

An important point is the matching of both air gaps at zero torque. Any offset would lead to unbalanced bridge and therefore to a temperature drift of the sensor zero point. Also, non equal air gap size over the circumference would lead to 2nd order effects causing small output changes with angular rotor position. Therefore it is proposed that the ferromagnetic sleeves are separated exactly after bonding by laser cutting.

As a conclusion it is obvious that this sensor concept is very robust and self-compensating for all kinds of relative positioning tolerances ranging from temperature change, bearing play and forces applied to the stator or rotor shaft.

3 SENSOR OPTIMIZATION

This chapter describes the basic kinematical principle and the FEM-optimization of the torque sensor geometry (Marik, 2008). Optimization criteria are the translational movement Δx, but also the costs for manufacturing and assembling.

3.1 Kinematical Principle

Basically the sensibility of the torque sensor depends on the translation Δx which is directly linked to the current torsion angle. The formal correlation of the torsion angle (φ), the torque (Mx), the length of the casing (l), the modulus of rigidity (G) and the polar area moment (Ip) is shown in equation (1).

$$\varphi = \frac{M_X * l}{G * I_P} * \frac{180}{\pi} \qquad (1)$$

In most cases the shaft diameter, the external torque, the material of the shaft and the possible casing length are given values and cannot be changed. Concerning the example of an 18mm diameter steering shaft, a torsion angle φ from 0.05 to 0.06 degrees leads to a twist movement on the shaft surface of 0.02mm. According to that point the only way to achieve a maximum Δx is to optimize the design of the mechanical sleeve.

The simplified function model of the sleeve is a two rod system with three revolute joints. Figure 4 shows the situation on a flat surface to demonstrate the geometrical relationships more easily.

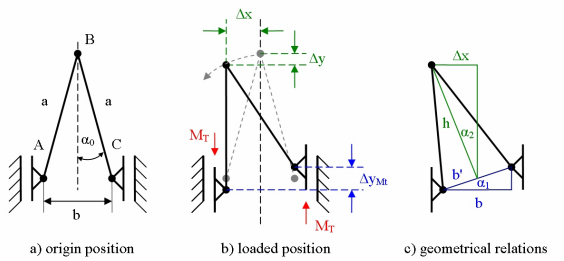


Figure 4: Simplified function model of the sleeve on a flat surface.

Two rods with the length (a) are connected to each other at joint (B) and to the surrounding at joints (A) and (C) (see Figure 4a). The connection points A and B are only moveable in the y-direction but fixed in the x-direction. When the system gets loaded due to a torque the points A and C move contrarily along the y-direction (ΔyMt) and hence the joint B moves from its origin position along a circular path with the radius (a) (see Figure 4b).

The geometrical relations in x-direction are shown in Figure 4c and can be described in equation (2).

$$\Delta x = \sqrt{a^2 - \frac{b^2 + \Delta y_{Mt}^2}{4}} * sin(tan^{-1}(\frac{\Delta y_{Mt}}{b})) \qquad (2)$$

From equation (2) it is obvious that the achieved Δx is linear depending on the length (a) and nonlinear to the geometric relation of ΔyM and the width (b), the tilt angle of the system. But when the two rod system gets wrapped around a cylinder surface the interacting rod length will be scaled down by the influence of the cylinder radius R as shown in Figure 5.

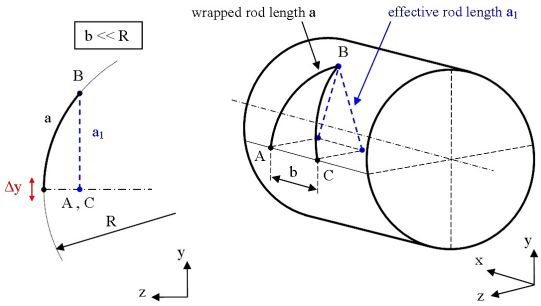


Figure 5: Function model on a cylindrical surface.

Consequently to that effect the linear impact of length (a) gets nonlinear according equation (3) under the precondition that the distance (b) is very short in relation to the radius (R) (b << R).

$$a_1 = R * sin(\frac{a}{R}) \qquad (3)$$

It is obvious that (a1) has its maximum value when (a) is the quarter of the cylinder circumference, which is an important fact for the geometry of the rotor rod elements.

3.2 FEM-Studies

Based on the kinematical principle the final CAD-design has to be defined. The optimized shape of the measuring sleeve (see Figure 1) is characterized by a maximum Δx and a light structure with high stiffness. Simulation helps to accelerate the development process and to increase the quality of the first prototype (Seiffert, 2008). The stresses in critical regions near the revolute joint positions must be under the fatigue limit of the chosen aluminium alloy. The prototype sleeve is to be made as a turning-milling part. Therefore the minimum cutting

tool diameter and the attainable lowness of the inner undercuts also have to be considered.

Figure 6 shows the displacement (dx) in the direction of the shaft axis. It is the result of a static FEM analysis with the external torque at one side and the fixation at the other side as model constraints.

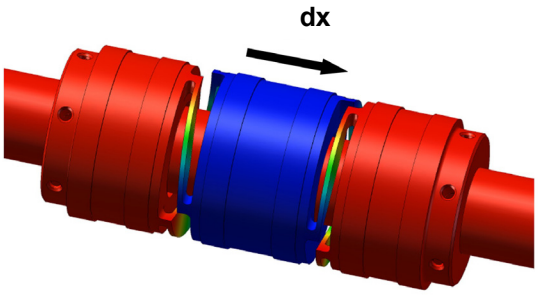


Figure 6: FEM result of displacement dx.

A satisfying compliance between measurement results and simulation has been found (see Figure 7). The measurement setup is illustrated in Figure 8.

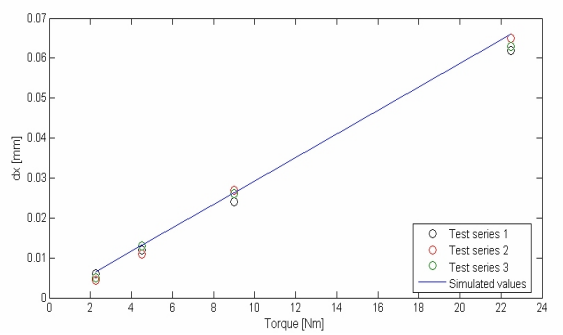


Figure 7: Comparison between measuring and simulation.

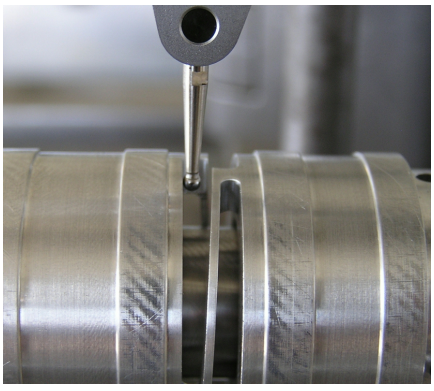


Figure 8: Measurement setup.

In addition theoretical considerations can be proved by reducing the dimensions of the joint connections. Figure 9 shows the differences between the results of the basic kinematic model and the FEM simulations by varying rod lengths (a). Figure 10 gives an overview about changes of (dx) with different values (b).

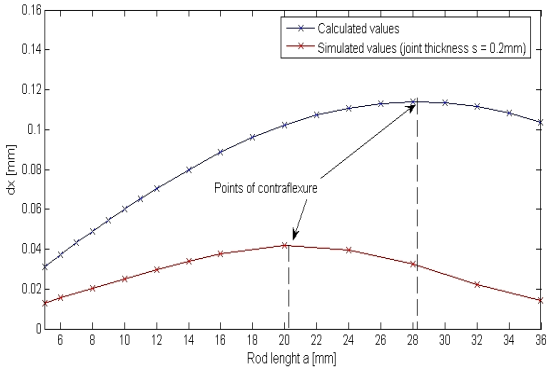


Figure 9: Displacement values with various rod lengths.

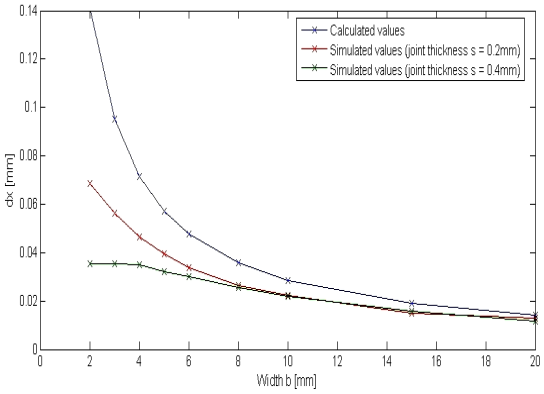


Figure 10: Displacement values with various values b.

4 INTERFACE ELECTRONICS

The inductive half bridge is supplemented by a resistive voltage divider to form a Maxwell bridge. Its output signal is amplified and converted to digital output with high precision by an integrated data acquisition circuit. The block diagram in Figure 11 shows the setup of the interface electronics.

Today's semiconductor circuit design techniques enable us to conquer new domains in precision and ultra low drift signal conditioning unreachable just a few years ago.

For this torque sensor an integrated data

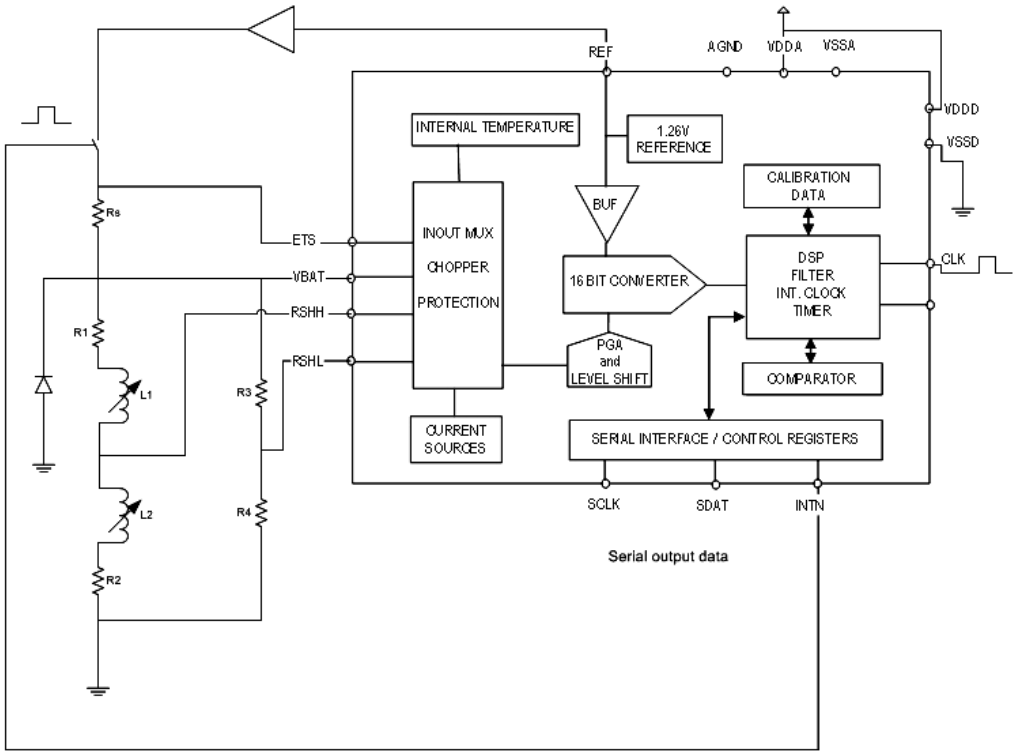


Figure 11: Block diagram of sensor interface electronics.

acquisition device was used which was specifically designed to convert very small voltages to digital domain with very high precision.

This data acquisition device offers two multiplexed fully differential input channels, a programmable gain amplifier, a 16 bit analog to digital converter and a serial interface for device configuration and result read-out.

Special auto zero offset architecture offers virtually zero offset below 0.5µV and very low noise. The analog to digital converter is of Sigma-Delta type, an architecture which offers high linearity without any missing codes.

The device has an integrated trim-able precision reference and a temperature compensation for the entire measurement path. The integrated circuit also offers a measurement-ready signal which is used to excite the transducer bridge with a pulsed DC voltage in conjunction with a diode which is to discharge the energy stored in the coils after each applied pulse.

The second channel can be used to measure the bridge current to compensate the temperature coefficient of the copper coil which would otherwise lead to a scaling error. If the digital representation of the bridge voltage is multiplied by compensation factor proportional to the bridge current in an external micro controller, the final output signal can be made independent from temperature. In that case the coil wires are acting as a temperature sensor.

The development of the electronic system was carried out at the Department of Automation Technology at FH *CAMPUS* 02 in collaboration with austriamicrosystems Corporation applying the newest chip generation (Pauritsch, 2008).

5 TEST SYSTEM

Based on the presented solution a test system has been developed that demonstrates a steering (see Figure 12). For that purpose the demonstrator consists of a steering shaft with torque sensor and rotary encoders, two bearings and a steering wheel.

5.1 Torque Sensor

The torque sensor is placed in the middle of the bearing houses. The sensor sleeve is pinned on the 18 mm diameter shaft to induce the shaft twist into the sleeve. The stator part consists of a casing with two coils and four ferromagnetic concentrators forming two inductances in the range of 10mH with

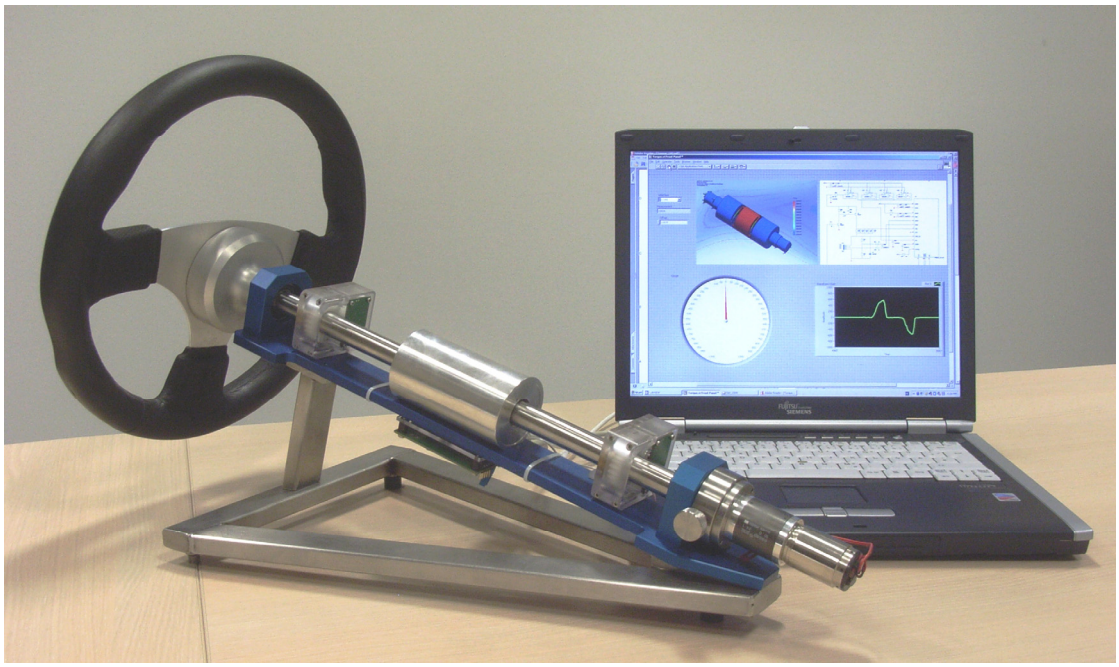


Figure 12: Steering demo system with one torque sensor, two rotary encoders and Graphical- User-Interface (GUI) for torque measurement.

400 windings of 0,15mm copper wire. The entire coil arrangement is fixed by two screwed covers.

The stator has to be aligned to the axis of the shaft carefully.

The coil wires are led through small holes and connected with the sensor interface. The PCB is mounted on the base plate of the demonstrator. The steering torque can be simulated by a DC-motor or by a special brake with a fixation screw.

5.2 Rotary Encoder

In addition to the torque measurement the steering angle is also measured at two positions by using Hall-sensors.

The rotary encoder (Czichos, 2008) consists of a small magnet disc with north/south pole, which rotates at a very short distance to the Hall-sensor-chip. This disc is directly mounted on a small gearwheel. The according gear consists of the pinion gear with the magnet and a gearwheel that is fixed on the steering shaft. The casing of the rotary encoder is separated into two parts for easy assembling and also includes the sensor PCB.

5.3 User Interface

The user interface software is programmed with "Labview" and enables various types of graphical representation of the measurement results.

5.4 Linearity Measurement

The characteristic of the new sensor is the result of measurements within a specified torque range of ±5Nm. Figure 13 shows the sensor output signal which is virtually linear with torque without any hysteresis.

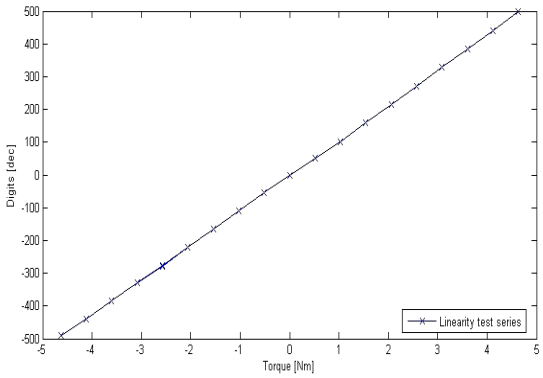


Figure 13: Torque sensor characteristic without any hysteresis.

6 CONCLUSIONS

A contactless torque sensor for accurate and reliable low frequency torque measurement on a rigid steering shaft has been introduced. The principle is in our opinion well suited to develop a low cost sensor product with adequate values of repeatability, accuracy and reliability. The cost and quality targets can be achieved, if manufacturing techniques like laser cutting and laser welding will be utilized.

The entire mechatronic solution is based on well known design elements which do not lead to long term drifts of zero point and output scale. There is virtually no hysteresis in output signal as proven by the demonstrator design.

It has to be pointed out that the design of a final sensor product has to consider sufficient spring stiffness in axial direction as well as low mass for the displaceable centre sleeve spring/mass system to ensure insignificant displacement when the sensor is exposed to low frequency or constant acceleration. It has to be ensured that its first mode resonance frequency is higher than the frequency band of interest.

Our contribution represents a preparatory study for any subsequent mechatronic product development.

Due to the number of involved technical disciplines like mechanical-, electromagnetic-, electronic-, manufacturing- and safety engineering it is recommended to continue the research work as an academic project in conjunction with other universities and industrial partners.

We are confident that as a result of such R&D-activities a new type of mechatronic torque sensor device with unmatched performance versus cost ratio can be placed on the market for many applications.

REFERENCES

Angleviel, D., Frachon, D., and Masson, G., 2006. *Development of a Contactless Hall effect torque sensor for Electric Power Steering*. MMT S.A.

Czichos, H., 2008. *Mechatronik.* P 217. Vieweg+Teubner Verlag. Wiesbaden.

Graßmann, O., Henrichfreise, H., Niessen H., and Hammel K., 2003. *Variable Lenkunterstützung für eine elektromechanische Servolenkung*. 23. Tagung "Elektronik im Kfz". Stuttgart.

Heißing, D., Ersoy M., 2008. *Fahrwerkhandbuch*. pp 209-215. Vieweg+Teubner Verlag. Wiesbaden.

Jacobsen, A.M., 1944. *Electrical Dynamometer*. US Patent Application Serial No. 561,467. Los Angeles.

Marik, R., 2008. *Konturoptimierung einer Messnabe.* Bachelor Thesis. FH CAMPUS 02. Graz.

Pauritsch, M., 2008. *Product documentation of the sensor interface*. Graz..

Seiffert, U., Rainer G., 2008. *Virtuelle Produktentstehung für Fahrzeug und Antrieb im Kfz*. pp 7-29. Vieweg+Teubner Verlag. Wiesbaden.

INDUCING COOPERATION IN FUZZY CLASSIFICATION RULES USING ITERATIVE RULE LEARNING AND RULE-WEIGHTING

Omid Dehzangi, Ehsan Younessian
Nanyang Technological Universit, Singapore
dehzangi@pmail.ntu.edu.sg, ehsa0001@ntu.edu.sg

Fariborz Hosseini Fard
SoundBuzz PTE LTD, Subsidiary of Motorola Inc., Singapore
cbfn87@motorola.com

Keywords: Fuzzy Systems, Classification, Iterative Rule Learning (IRL), Rule Weighting, ROC.

Abstract: Fuzzy Rule-Based Classification Systems (FRBCSs) focus on generating a compact rule-base from numerical input data for classification purposes. Iterative Rule Learning (IRL) has been proposed to reduce the search space for learning a rule-set for a specific classification problem. In this approach, a rule-set is constructed by searching for an appropriate fuzzy rule and adding it to the rule-set in each iteration. A major element of this approach is the requirement of an evaluation metric to find the best rule in each iteration. The difficulty in choosing the best rule is that the evaluation metric should be able to measure the degree of cooperation of the candidate rule with the rules found so far. This poses a major difficulty when dealing with fuzzy rules; because unlike crisp rules, each pattern is compatible with a fuzzy rule only to a certain degree. In this paper, the cooperation degree of a candidate rule is divided into the following two components: I)- The cooperation degree of the rule with other rules of the same class, II)- The cooperation degree of the rule with rules of the other classes. An IRL scheme to generate fuzzy classification rules is proposed that induces cooperation among the rules of the same class. Cooperation between the rules of different classes is handled using our proposed rule-weighting mechanism. Through a set of experiments on some benchmark data sets from UCI-ML repository, the effectiveness of the proposed scheme is shown.

1 INTRODUCTION

The main application area of fuzzy rule-based systems has been control problems (Sugeno, 1985). Fuzzy rule-based systems for control problems can be viewed as approximators of nonlinear mappings from non-fuzzy input vectors to non-fuzzy output values. Recently, fuzzy rule-based systems have often been applied to classification problems where non-fuzzy input vectors are to be assigned to one of a given set of classes. Many approaches have been proposed for generating and learning fuzzy if-then rules from numerical data for classification problems. For instance, FRBCSs are created by simple heuristic procedures (Ishibuchi et al., 1992), (Abe, 1995), neuro-fuzzy techniques (Nauck and R. Kruse, 1997), clustering methods (Abe and Thawonmas, 1997), genetic algorithms (Ishibuchi et al., 2005), etc.

Pattern classification has been the main issue in machine learning. Classification is to acquire knowledge from a set of training patterns and use this knowledge to predict the class of a new pattern. FRBCSs use fuzzy rules as a mean to perform classification tasks. A rule is an if-then relation from the n-dimensional pattern space to the set of classes. In a single winner rule approach (Ishibuchi and Nakashima, 2001), to classify an unknown pattern, one rule is selected and used to classify the pattern. In this paper, a single winner rule approach is used which will be discussed later. In the broadest sense, any method that incorporates information from training samples in the design of a classifier employs learning. Therefore, designing classifiers involves some type of learning to learn or estimate unknown parameters using a set of labeled patterns.

In this paper, an IRL approach for fuzzy rule selection is presented in which the degree of cooperation of each candidate rule with other rules of the same class is estimated. In this approach, the final rule-set for classification is constructed by searching for an appropriate fuzzy rule and adding it to the rule-set in each step. Then, a simple rule-weighting mechanism is proposed to reach some degrees of cooperation/competition among the rules of different classes. Four UCI ML driven data sets are then used to evaluate the proposed fuzzy classification method.

2 FUZZY CLASSIFICATION RULES

In the design of fuzzy rule-based systems, we face two conflicting objectives: error minimization and interpretability maximization. Error minimization has been used in many applications of fuzzy rule-based systems in the literature while the interpretability was not usually taken into account in those applications. Recently, the tradeoff between these two objectives has been discussed in some studies. When fuzzy rule-based systems are used for two-dimensional problems, fuzzy rules can be represented in a tabular form (Ishibuchi and Yamamoto, 2004). Figure 1 shows an example of a fuzzy rule table for a two-dimensional pattern classification problem. In this figure, we have the following four fuzzy rules:

If X_1 is *small* and X_2 is *small* then Class 1,
If X_1 is *small* and X_2 is *large* then Class 2,
If X_1 is *large* and X_2 is *small* then Class 3,
If X_1 is *large* and X_2 is *large* then Class 4,

where *small* and *large* are linguistic values defined by triangular membership functions.

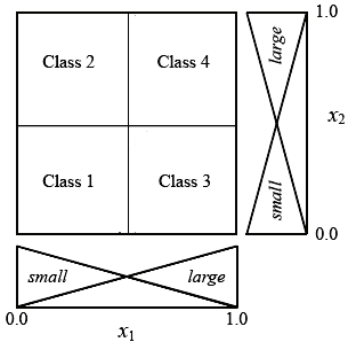


Figure 1: Four fuzzy rules in the 2-dimensional pattern space $[0,1]\times[0,1]$.

As shown in Figure 1, fuzzy rules for 2-dimensional problems can be written in a human understandable manner using the tabular form representation. When fuzzy rule-based systems are applied to high-dimensional problems, their interpretability is significantly degraded due to the two difficulties: the increase in the number of fuzzy rules and the increase in the number of antecedent conditions of each fuzzy rule.

Assume that we have m labeled patterns $X_p=(x_{p1},\ldots,x_{pn})$, $p=1,2,\ldots,m$ from M classes in an n-dimensional continuous pattern space is given. For classification problems with n number of attributes, as in (Ishibuchi and Yamamoto, 2004), we use fuzzy rules of the following form:

$$\text{Rule } R_i: \text{If } x_1 \text{ is } A_{i1} \text{ and } \ldots \text{ and } x_n \text{ is } A_{in} \text{ then class } C_i \text{ with } CF_i \quad (1)$$

where R_i is the i-th rule, $X=(x_1,\ldots,x_n)$ is an n-dimensional pattern vector, A_{ij} is an antecedent fuzzy set (i.e., linguistic value such as small or large in Figure 1), C_i is the class label of R_i, and CF_i is the weight of R_i. It should be noted that the consequent part of our fuzzy rule for classification problems is totally different from standard fuzzy rules for function approximation problems. The consequent of our fuzzy rule is a non-fuzzy class label and the rule weight CF_i is a real number in the unit interval [0, 1]. The rule weight is used as the strength of each fuzzy rule when a new pattern is classified by a fuzzy rule-based classification system (see (Ishibuchi and Nakashima, 2001) for details).

The compatibility grade of a training pattern X_p with the antecedent part $A_i=(A_{i1},\ldots,A_{i,n})$ of fuzzy rule R_i is calculated using product operator as,

$$\mu_{A_i}(X_p)=\mu_{A_{i1}}(X_{p1})\times\cdots\times\mu_{A_{in}}(X_{pn}) \quad (2)$$

where $\mu_{A_{ij}}(\cdot)$ is the membership function of the antecedent fuzzy set A_{ij}.

3 CANDIDATE RULE GENERATION

In our approach, fuzzy if-then rules are generated from numerical data. Then, the generated rules are used as candidate rules from which a small number of fuzzy if-then rules are selected in an iterative manner. The domain interval of each attribute x_i is discretized into K_i fuzzy sets. Figure 2 shows some examples of fuzzy discretization.

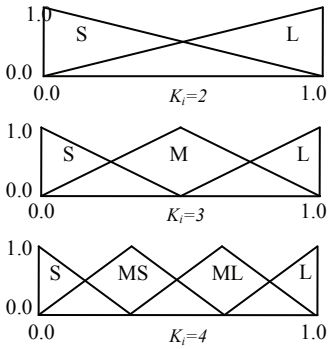


Figure 2: Some typical examples of fuzzy partitions of the domain interval [0, 1].

The meaning of each label is as follows:
S: small, MS: medium small, M: medium, ML: medium large, and L: large. The superscript of each label denotes the granularity of the corresponding fuzzy partition.

Each antecedent fuzzy set in a fuzzy rule can be one of K_i fuzzy sets or "*don't care*". Therefore the total number of possible antecedent combinations is $(K_1+1)\times\ldots\times(K_n+1)$.

To determine the consequent part of a rule, we use a concept in data mining called confidence degree. The confidence of a fuzzy association rule is defined as (Ishibuchi and Yamamoto, 2004):

$$c(A_i \Rightarrow class\,h) = \frac{\sum_{x_p \in class\,h} \mu_{A_i}(X_p)}{\sum_{p=1}^{m} \mu_{A_i}(X_p)} \quad (3)$$

The consequent class C_i of the fuzzy rule R_i is specified by identifying the class with the maximum confidence. If the maximum confidence of a rule is zero or the difference between the first and second maximum confidences is zero, the rule is not generated.

To avoid coping with a large number of candidate rules in the rule selection procedure, some prescreening criterion is needed. Several criteria is used in the previous works (Gonzalez and Perez, 1999). In this paper we use the following criterion:

$$value(A_i \Rightarrow \text{class } h) = \sum_{x_p \in class\,h} \mu_{A_i}(X_p) - \sum_{x_p \notin class\,h} \mu_{A_i}(X_p) \quad (4)$$

4 RULE SELECTION

After generating the candidate rules, a set of rules must be selected to construct the rule-base of the classifier. The rules are selected in an iterative manner. The generated fuzzy if-then rules are divided into M groups according to their consequent classes. Fuzzy if-then rules in each group are sorted in descending order of the evaluation criterion (4).

In the first step of the rule selection the best rule of each class is added to the rule-base. To build a rule-base with N rules ($N \geq M$), the remaining N-M rules are selected one by one. A major element of this approach is the need of an evaluation metric to find the best rule in each iteration.

The difficulty in choosing the best rule is that the evaluation metric should be able to measure the degree of cooperation of the candidate rule with the rules found so far. This is a major difficulty when dealing with fuzzy rules, due to the fact that each pattern is compatible with a fuzzy rule to a certain degree.

For the rules found so far, a measure called "*fuzzy accuracy measure*" of the rule-base is defined as:

$$F_{rule-base} = \sum_{x_p \in \text{class } h} \max_{R_i \in rule-base}\left(\mu_{R_i}(x_p)\right) - \sum_{x_p \notin \text{class } h} \max_{R_i \in rule-base}\left(\mu_{R_i}(x_p)\right) \quad (5)$$

The aim of this measure is to calculate the overall effectiveness of the rules of the same class that are found so far. To add the rule R_w from the set of candidate rules to the rule-base, the rule that improves $F_{rule\text{-}base}$ the most is chosen:

$$R_w = \arg\max_{R_i \in Candidate\_Rules} \left\langle F_{rule-base \cup \{R_j\}} - F_{rule-base} \right\rangle \quad (6)$$

The process of rule selection is continued iteratively as long as there are further improvements in $F_{rule\text{-}base}$. The proposed scheme both induces cooperation among the rules of the same class and avoids including redundant rules in the final rule-base which results in having a compact rule-base.

5 INDUCTING COOPERATION WITH RULE WEIGHTING

The first component of the cooperation of the newly added rule is its degree of cooperation with the rules of the same class. This component is considered in the rule selection phase. The second component of the cooperation is the degree of cooperation between rules of different classes. This component is also called *competition*. Competition among the rules of different classes is handled by assigning a weight to each different rule.

In (Nauck & Kruse, 1998), the effect of rule weights in fuzzy rule-based systems for function

approximation problems is discussed. They also showed how the modification of the membership functions of antecedent or consequent fuzzy sets can be equivalently replaced by the learning of rule weights. Several heuristic criteria for rule-weighting have been introduced in earlier works done by Ishibuchi et al (Ishibuchi and Yamamoto, 2004) which are briefed here:

$$CF_1 = c(A_i \Rightarrow C_i) \tag{7}$$

$$CF_2 = c(A_i \Rightarrow C_i) - \frac{1}{M}\sum_{\substack{t=1 \\ t \neq C_i}}^{M} c(A_i \Rightarrow C_i) \tag{8}$$

$$CF_3 = c(A_i \Rightarrow C_i) - \max\left\{c(A_i \Rightarrow Class\, t \,|\, t = 1,2,...M; t \neq C_i)\right\} \tag{9}$$

$$CF_4 = c(A_i \Rightarrow C_i) - \frac{\sum_{x_p \notin Class\, C_i} \mu_{R_i}(x_p)}{\sum_{x_p} \mu_{R_i}(x_p)} \tag{10}$$

where $c(A_i \Rightarrow C_i)$ is the confidence of a fuzzy rule R_i, and $\mu_{R_i}(X_p)$ is the compatibility grade of a training pattern X_p with the antecedent part of fuzzy rule R_i.

In the following, a simple rule-weighting criterion is presented. In our suggested method, it is tried to reach some degrees of cooperation /competition among the rules of different classes. To calculate the weight of the fuzzy rule R_i, first a value is calculated named as *contrast* for each training data point X_p:

$$Contrast_{R_i}(X_p) = \frac{\max\limits_{\substack{j=1,...,N \\ label(R_j) \neq C_i}} \left(\mu_{R_j}(X_p)\right)}{\mu_{R_i}(X_p) + \max\limits_{\substack{j=1,...,N \\ label(R_j) \neq C_i}} \left(\mu_{R_j}(X_p)\right)}, \tag{11}$$

where R_i is the rule that is being weighted. If a data point is covered by the rules of other classes, the *contrast* value of this data point, with respect to the rule in hand, is close to one; otherwise it is closer to zero.

Data points are sorted in ascending order of their *contrast* values. The next step is to find a threshold of the *contrast* values, ω, that best separates the data points of the same class from the data points of other classes. In this way, each data point X_p for which $Contrast_{R_i}(X_p) < \omega$ is assumed to be of the same class as R_i. The threshold is then altered from the list *contrast* value to the greatest and accuracy of the classifier with respect to the current threshold is measured. The weight of rule R_i is obtained from the the value of the best threshold (i.e. leading to the highest accuracy) normalized in the range of [0, 1] as follows,

$$CF_i = \frac{\omega}{1+\omega} \tag{12}$$

6 EXPERIMENTAL RESULTS

In our experiments, we used four data sets in Table 1 available from the UCI ML repository (Merz and Murphy, 1996).

Table 1: Statistics of the data sets used in our experiments.

Data set	# of attributes	# of patterns	# of Classes
Pima	8	768	2
Wine	13	178	3
Cancer Wis.	9	699	2
Glass	9	214	6

All attribute values of the four data sets were normalized into real numbers in the unit interval [0, 1] before extracting fuzzy rules. Since we did not know an appropriate fuzzy partition for each attribute of each test problem, we simultaneously used three different fuzzy partitions in Figure 2. One of the 9 triangular fuzzy sets was used as an antecedent fuzzy set. To generate simple fuzzy rules (i.e., short fuzzy rules with a small number of antecedent conditions), we also used "*don't care*" as an antecedent fuzzy set. The membership function of "*don't care*" is defined as $\mu_{"dontcare"}(X) = 1$. The total number of combinations of antecedent fuzzy sets is 10^n for an n-dimensional problem.

In our computational experiments we only examined fuzzy rules with three or less antecedent conditions (i.e., with n-3 or more "*don't care*" conditions). The restriction on the number of antecedent conditions is to generated interpretable fuzzy rules as well as for decreasing the CPU time.

In Tables 2-5, the results of the fuzzy classification system using the proposed fuzzy rule selection method with different rule-weighting methods are shown on the data sets of Table 1. All the reported results are the average of ten trials of ten-fold cross validation. The first column of each Table is the number of rules used to classify the data points in the selected data set. The other five columns represent the classification accuracy of the

four mentioned weighting methods proposed in (Ishibuchi and Yamamoto, 2004) compared to our proposed method. As it can be seen in the results, the proposed method led to the best results among the rule-weighting methods. In each row of the Table 2-5, the method which had the best result is bolded.

Table 2: Test data classification rates of Glass dataset.

# of rules	No Weight	CF1	CF2	CF3	CF4	Our Method
6	49.61	48.95	49.42	48.71	54.15	**56.99**
12	55.72	58.03	58.80	60.56	60.37	**63.89**
18	57.81	59.43	58.37	59.41	63.08	**66.29**
24	61.25	60.85	63.18	60.47	62.38	**67.18**
30	61.35	63.05	61.47	61.33	63.78	**67.47**
36	62.08	62.37	63.68	62.13	65.51	**68.11**
42	61.21	60.28	61.75	63.53	64.18	**68.29**
45	62.98	61.63	63.14	64.22	65.01	**68.62**

Table 3: Test data classification rates of Wine dataset.

# of rules	No Weight	CF1	CF2	CF3	CF4	Our Method
3	84.90	87.27	87.82	86.99	85.97	**85.54**
6	91.55	92.53	93.31	91.69	91.85	**93.14**
9	93.14	91.89	92.28	92.86	94.14	**91.97**
12	92.88	94.81	94.96	93.77	93.38	**92.11**
15	93.94	93.16	94.84	94.69	93.44	**95.51**
18	94.57	93.86	93.78	93.60	92.73	**95.48**
51	95.33	95.00	94.56	94.64	93.34	**95.60**
56	95.18	94.37	94.66	94.53	94.42	**95.64**

Table 4: data classification rates of Cancer dataset.

# of rules	No Weight	CF1	CF2	CF3	CF4	Our Method
2	81.84	83.29	80.81	81.06	**83.16**	83.13
3	91.79	91.25	91.65	**92.67**	**92.04**	91.16
4	89.61	91.41	92.34	**92.44**	92.36	91.61
5	92.87	91.34	90.35	90.57	**93.08**	92.20
6	93.16	**93.66**	93.32	92.55	92.59	90.81
9	90.44	94.55	91.98	91.00	91.14	**94.82**
12	92.66	92.87	90.70	91.63	92.60	**94.91**
17	93.49	91.66	92.34	92.25	91.73	**95.44**

Table 5: Test data classification rates of Pima dataset.

# of rules	No Weight	CF1	CF2	CF3	CF4	Our Method
2	68.53	69.80	69.34	68.10	69.20	**68.85**
5	69.1	71.66	68.64	70.22	68.22	**73.64**
7	68.15	70.28	71.05	69.20	70.34	**76.03**
10	70.52	69.59	68.47	70.52	70.38	**74.23**
18	71.79	70.08	70.59	70.36	70.49	**74.92**
27	73.11	70.1	70.73	70.46	70.99	**75.24**
37	71.40	70.53	72.32	71.67	70.39	**75.78**
50	70.97	72.56	71.32	71.86	71.47	**76.22**

Although the classification accuracy has always been the main concern in classification problems, interpretability also have to be considered. There are two factors that heavily affect the interpretability of a rule-based system: number of the generated rules and number of antecedent conditions of each generated rule. As shown, our proposed method is highly interpretable in terms of both number the generated fuzzy classification rules and their number of antecedent conditions.

In Table 6, we compared our results to the results obtained by another successful rule-based method as benchmark results called C4.5 reported by (Elomaa and Rousu, 1999). As shown in Table 6, except in one case, the proposed classifier in this paper shows higher classification rates.

Table 6: Accuracy of the proposed classifier compared to C4.5. The best result in each row is highlighted by boldface.

Data set	The proposed classifier (%)	C4.5 classifier	
		Worst (%)	Best (%)
Pima	**76.2**	72.8	75.0
Cancer	**95.4**	94.0	94.9
Wine	**95.6**	92.2	94.4
Glass	68.6	68.8	**72.7**

7 CONCLUSIONS

In this paper, the cooperation degree of the fuzzy classification rules was divided into the two components: I)- The cooperation degree of the rules with other rules of the same class, II)- The cooperation degree of the rules with rules of the other classes. We proposed an IRL method for fuzzy rule selection. Using the proposed criterion, it was possible to estimate the degree of cooperation of a candidate rule with other rules of the same class in

the final rule-base. Furthermore, a simple rule-weighting mechanism was proposed to reach some degrees of cooperation/competition among the rules of different classes. The experimental results on real problems like speech data classification showed the effectiveness of the proposed method to generate fuzzy classification rules with high degrees of cooperation among them.

REFERENCES

Ishibuchi H., Nakashima T., 2001. Effect of Rule Weights in Fuzzy Rule-Based Classification Systems. *IEEE Transaction on Fuzzy Systems*.

Sugeno M., 1985. An introductory survey of fuzzy control, *Information Sciences*, 36: 59-83.

Ishibuchi H., Nozaki K., Tanaka H., 1992. Distributed representation of fuzzy rules and its application topattern classification. *Fuzzy Sets Systems*, 52: 21-32.

Abe S., Lan M., 1995. A method for fuzzy rules extraction directly from numerical data and its applicationto pattern classification. *IEEE Trans. on Fuzzy Systems*, 3: 18-28.

Nauck D., Kruse R., 1997. A neuro-fuzzy method to learn fuzzy classification rules from data. *Fuzzy Sets and Systems*, 89: 277-288.

Abe S., Thawonmas R., 1997. A fuzzy classifier with ellipsoidal regions. *IEEE Trans. on Fuzzy Systems*, 5: 358-368.

Ishibuchi, H., Yamamoto, T., Nakashima, T., 2005. Hybridization of Fuzzy GBML Approaches for Pattern Classification Problems, *IEEE Transaction on Systems, Man, and Cybernatics*.

Gonzalez, A., Perez, R., 1999. SLAVE: A genetic learning system based on an iterative approach, *IEEE Trans. on Fuzzy Systems*, 7: 176-191.

Ishibuchi H., Yamamoto T., 2004. *Comparison of Heuristic Criteria for Fuzzy Rule Selection in Classification Problems.* Kluwer Academic Publishers.

Nauck, D., Kruse, R., 1998. How the learning of rule weights affects the interpretability of fuzzy systems," *Proc. of 7th IEEE International Conference on Fuzzy Systems*, 1235-1240.

Ishibuchi H., Yamamoto T., 2003. Effects of Three-Objective Genetic Rule Selection on the Generalization Ability of Fuzzy Rule-based Systems, *The Genetic and Evolutionary Computation Conference*.

Merz, C.J., Murphy, P.M., 1996. *UCIRepository of Machine Learning Databases*. Irvine, CA: University of California Irvine, Department of information and Computer Science. Internet:
http://www.ics.uci.edu/~mlearn/MLRepository.html

Elomaa, T., Rousu, J., 1999. General and efficient multisplitting of numerical attributes, *Machine Learning* 36: 201-244.

A HYBRID METAHEURISTIC FOR SOLVING SINGLE MACHINE SCHEDULING PROBLEM

Adrian Serbencu, Viorel Mînzu, Daniela Cernega and Adriana Serbencu
Control Systems and Industrial Informatics Department "Dunarea de Jos"Galati University
Domneasca – 47, Galati, Romania
Adrian.Serbencu@ugal.ro, Viorel.Minzu@ugal.ro, Daniela.Cernega@ugal.ro, Adriana.Serbencu@ugal.ro

Keywords: Discrete optimization, Manufacturing, Metaheuristics, Stochastic descent, Ant Colony Systems.

Abstract: This paper proposes a metaheuristic for solving the Single Machine Scheduling Problem that is implemented by a hybrid system made up of an Ant Colony System and a stochastic descent algorithm called Kangaroo. The hybrid system is based on the collaboration between a social type multiagent system and an Iterated Solution Improvement method.

1 INTRODUCTION

The main purpose of multiagent systems is the distributed solving of problems. A special type of problem, which can be solved in a distributed way, is the combinatorial optimization problem. The idea of the algorithm "ant system" (Dorigo, *et al.,* 1996) has the source in the study of insects collective behavior. The ants have the capability to act together in order to perform a task, but any of them could not perform alone the task (Beckers, *et al.,* 1992). This is a distributed solving mechanism because every agent has only a very small contribution. The complex collective behavior and the interactions between agents are fundamental in the field of artificial life.

This paper proposes a metaheuristic for solving the Single Machine Scheduling Problem (SMSP). For a given processor and a set of jobs that must be executed on this processor, the problem is to determine the sequence of jobs such that the weighted tardiness (defined in section 2) is minimized. Obviously, because of the combinatorial aspect, this kind of problem is NP-complete. Hence, sub-optimal solutions are generally preferred to optimal ones. A sub-optimal solution is given by an approximation algorithm like genetic algorithm, simulated annealing, tabu search, stochastic descent algorithms, etc. In paper (Madureira, *et al.*, 2000), an interesting practical resolution is given, in the context of a scheduling system for Dynamic Single Machine Problem. The SMSP is solved using a genetic algorithm and thus, good results are obtained.

In exchange, algorithms like simulated annealing (Kirkpatrick, *et al.*, 1983), tabu search (Gloverf, 1989), stochastic descent (Papadimitriou and Steiglitz, 1982), etc. are *Iterated Solution Improvement* methods, which means that only one solution is improved by an iterative procedure. This kind of methods has abilities to intensify the local search and to detect the local minima. In the last years, hybrid metaheuristics (Vaessens, *et al,* 1992; Taillard *et al.*, 1998; Mahfoud, and Goldberg, 1995) have been developed, giving very interesting results. That is why, this paper proposes an Ant Colony System (ACS) based metaheuristic**,** described in section 3, formed by an ACS and a parallel version of a stochastic descent algorithm, called Kangaroo. The system has the collaborative power of the ACS and the intensification ability of the Kangaroo algorithm (KA).

The paper is organized as follows. In section 2, the Single Machine Scheduling Problem is stated. The general structure of the proposed hybrid system is presented in section 3 and the particularities of an ACS solving the SMSP are described in section 4. Section 5 outlines the implementation of the Kangaroo Algorithm, whereas the computational results are presented in section 6.

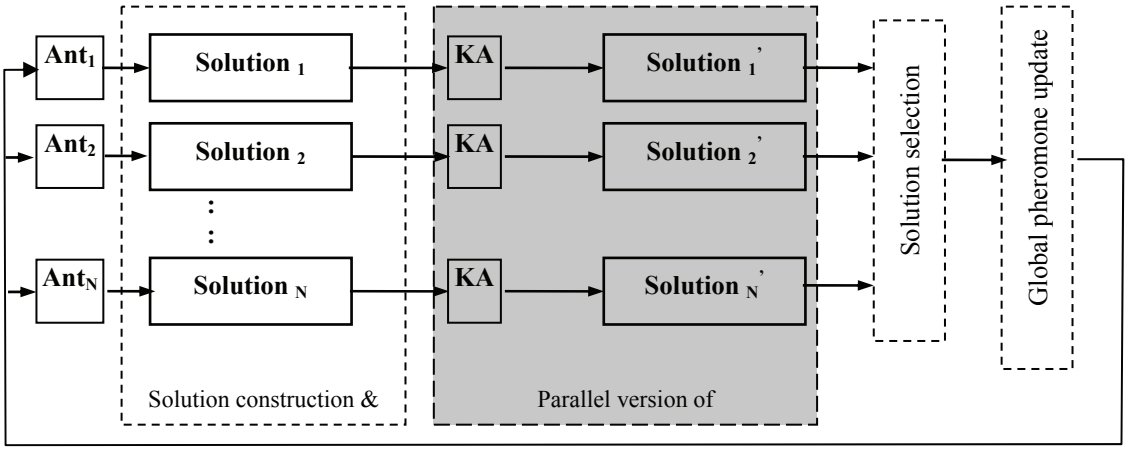


Figure 1: Ant Colony System with Kangaroo algorithm.

2 SINGLE MACHINE SCHEDULING PROBLEM

In order to minimize the total Weighted Tardiness for the SMSP the following assumptions are considered: a set of *n* independent jobs (*j= 1,…,n*) is available for processing at time zero and the attributes of the jobs are known in advance. The machine is never kept idle if there are any jobs to complete and it can handle only one job at a time. It processes the jobs without pre-emption. The jobs' set-up times are independent of the jobs' sequence, being included in the job processing times.

For each job *j*, it is considered p_j the processing time, d_j the due date, that means the date when the job should be completed, and w_j the penalty liable for each unit of delay. The jobs' completion starts at time *t*=0. The tardiness of a job is given by

$$T_j = Max\left\{t_j + p_j - d_j, 0\right\},$$

where t_j is the start time of job *j*. The objective function, which will be referred as fitness function, is:

$$f(u) = \sum_{j=1}^{n} w_j T_j$$

where *u* is a solution of the problem, that is a permutation of the job set:

$$u = [j_1, j_2, ..., j_i, j_{i+1}, ..., j_n],\ j_i \in \{1, 2, ..., n\}$$

The optimality criterion of the SMSP is $\min_u f(u)$.

This problem is a combinatorial optimization NP-complete problem. The problem can't be solved with deterministic optimal algorithms, as they require a computational time that increases exponentially with the problem size (Garey, and Johnson, 1979).

3 AN ANT COLONY SYSTEM BASED METAHEURISTIC

In papers (Bauer, *et al.*, 1999; Matthijs, *et al.*, 2000) the Ant Colony System is used to solve SMSP. In this approach, the artificial ants are constructing solutions for this problem and afterwards these solutions are considered initial solutions for a local optimization procedure.

The main idea of the proposed metaheuristic is to use a stochastic descent method instead of the local optimization procedure. In fact, this method is an Iterated Solution Improvement metaheuristic called Kangaroo. As a result, we have a special hybrid metaheuristic based on the collaboration between a social multiagent system - Ant Colony System - and a parallel version of Kangaroo algorithm.

The ACS is made up of N artificial ants which are constructing solutions of the optimization problem. The ants communicate using structured variables whose values represent a provisional quotation of the solutions or of parts of solutions quality. The value of these structured variables simulates the "pheromone" allowing the communication between ants in natural systems. In this case, the structured variables are grouped in a "pheromone" matrix.

The general optimization system is an iterative searching process for a better solution of the given problem. An iteration has two succesive phases. In

the first phase, ACS is constructing N solutions and the "pheromone" is continuously updated. The resulting solutions are taken over in the second phase, in order to be improved, by N instances of the KA. From the beginning, the KA makes a local optimization using the solutions produced by each ant as initial solutions. The KA is not limited to this single action, but it keeps trying to improve the current solution following its own strategie. Every instance of the KA is also an iterative procedure looking for a better solution than the current one, in a prescribed number of iterrations. The best found solution is selected and it will be used in a new global "pheromone" updating phase.

In the next iterration, the informations accumulated in the "pheromone" matrix will be used by the ACS, to guide the construction of the new set of solutions. Here after, an outline of the proposed metaheuristic is presented.

Do
1. The Ant Colony System constructs N solutions for the given problem;
2. The parallel version of Kangaroo Algorithm uses the N solutions from the first step as initial solutions and begins N stochastic descent processes. The result is a set of N other better solutions;
3. Select the best solution from this set and use it to update the "pheromone" matrix;

until the *stop criterion*.

Figure 2: General structure of the proposed metaheuristic.

As mentioned before, at the first step, the "pheromone" matrix is also updated during the solutions construction (see section 4). The stop criterion is usually a certain number of iterations.

4 ANT COLONY SYSTEM FOR SOLVING SMSP

Generally speaking, for solving a combinatorial optimization problem ACS needs two kinds of information(Dorigo, *et al.*, 1996; Dorigo, and Gambardella, 1997a; Dorigo, and Gambardella, 1997b). One of them is the heuristic information and the other one is specific to the ACS and concerns the "pheromone".

Each ant of ACS produces a solution of the problem, in the step 1 of the algorithm. This solution is a complete sequence of jobs obtained by an iterative process of placing a job j on the position i (Matthijs, *et al.*, 2000). At the position i, the ant chooses the job j meeting two constraints:
a) the job j is not already placed in the sequence and
b) the "pheromone" $\tau(i, j)$ has the maximum value for the job j.

For SMSP, the "pheromone" $\tau(i,j)$ is a quotation of the interest to place a job j on the position i. The heuristic information considered by an ant aiming to select a job for the current position may be represented by the inverse of the due date, or the Modified Due Date (MDD)(Bauer, *et al.*, 1999) computed with the formula

$$mdd_j = \max\{d_j, (C + p_j)\},$$

where C is the total processing time of the jobs already placed.

An ant k will select with probability q_0 the most attractive job, in order to be placed in the current position i, that is the task j assuring the maximum of $[\tau(i,j)]\cdot[\eta(i,j)]^{\beta}$. Nevertheless, the same ant may choose with the complementary probability (1-q_0) a job j using the probabilistic rule

$$p_k(i,j) = \begin{cases} \dfrac{[\tau(i,j)]\cdot[\eta(i,j)]^{\beta}}{\sum_{u\in J_k(p)}[\tau(i,j)]\cdot[\eta(i,j)]^{\beta}} & \text{if } q \in J_k(i) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where:
- q_0 is a parameter of the algorithm
- $\eta(i,j) = \dfrac{1}{d_j}$ or $\eta(i,j) = \dfrac{1}{mdd_j}$ is the heuristic information
- $J_k(i)$ is the set of not yet placed job by ant k;
- β is a parameter which determines the relative importance of heuristic information ($\beta > 0$).

When all the artificial ants have constructed the solutions, the algorithm uses the following rule for the global updating of the pheromone matrix:

$$\tau(i,j) \leftarrow (1-\alpha)\cdot\tau(i,j) + \alpha\cdot\Delta\tau(i,j) \quad (2)$$

where

- $\Delta\tau(i,j) = \begin{cases} T^{-1} & \text{if } (i,j) \in \text{best sequence} \\ 0 & \text{otherwise} \end{cases}$
- $0<\alpha<1$ is the pheromone increase parameter
- T is the total weighted tardiness of the global-best solution from the beginning of the trial.

The application of the rule (2) enforces only the

pheromone belonging to the best solution encountered until the current iteration. This is the reason of a premature convergence of the algorithm. Therefore, the rule of local pheromone updating is used. Every time the pheromone information is used by an ant selecting the job j for the position i, the rule (3) modeling the natural process of pheromone evaporation (forgetting) is applied:

$$\tau(i,j) \leftarrow (1-\rho)\cdot\tau(i,j) + \rho\cdot\Delta\tau(i,j) \qquad (3)$$

where $0<\rho<1$ is a parameter.

The term $\Delta\tau(i,j)$ in our implementation is set to τ_0, the initial pheromone level. In order to calculate τ_0, our algorithm constructs an initial solution for SMSP. In this solution, the jobs are placed in increasing order of the due date. The value of τ_0 is initialized with the inverse of the total weighted tardiness of this solution.

5 KANGAROO ALGORITHM

The KA is an approximation technique based on stochastic descent (Fleury, 1995), inspired by the simulation annealing method, but having a quite different searching strategy.

The "Kangaroo" method is implemented by an iterative procedure which minimizes an objective function f(u). A current solution u of the considered problem is replaced by a better one, situated in its neighborhood N(u), using a random selection. The algorithm tries "A" times to improve the current solution, where A is a parameter of the algorithm. If a new improvement is no longer possible, a "*jump*" procedure is performed, in order to escape from the attraction of a local minimum. This time the improvement of the current solution is not compulsory. This procedure can use a different neighborhood definition N'(u).

A detailed description of the KA is given in Minzu, and Henrioud, 1998. The stop criterion is either a maximum iteration number or a bottom bound of the objective function.

The best solution u^* encountered in the iterative process is memorized. At the end of KA, u^* is the "optimal" solution proposed by the algorithm.

The neighborhood N(u) is the set of solution u' obtained from u by the permutation of the jobs placed on positions i and $i+1$. For example, if u=[1 4 3 2 5], it holds N(u)={[4 1 3 2 5], [1 3 4 2 5], [1 4 2 3 5], [1 4 3 5 2], [5 4 3 2 1]}.

When a new improvement of the current solution is no longer possible u is replaced by a solution u' given by the *"jump"* procedure. In the case of SMSP a possible definition of the neighborhood N'(u) is the whole search space, but the KA converges (with probability 1) slowly to the global optimum. A very important aspect is the fact that deterministic heuristics may be integrated in *"jump"* procedure, in order to guide the search of an optimum solution keeping the convergence of the KA if the accessibility constraint is met (Minzu, and Henrioud, 1998). That is why, in the case of SMSP the neighborhood N'(u) may be the set of solution u' obtained from u by permutation of the job placed on i_{max} position, where i_{max} is the position in u of the job j_{max} where $j_{max} = \arg\max_{j=1,..,n} w_j T_j$, with a job placed on position $i<i_{max}$. So, the "*jump*" procedure determines the job with the biggest weighted tardiness and replaces it with a job situated on its left. In this way there is a chance to diminish the value of the criterion f(u).

6 IMPLEMENTATION AND COMPUTATIONAL RESULTS

In order to reduce the run time of the hybrid system, the job selection rule is applied on a reduced candidate list that does not contain all the unplaced

Table 1: Computational results of the stand-aloneACS.

Problem	Optimal value	Best value	Deviation of the Best Value %
Wt100-1	5988	8795	47
Wt100-2	6170	7724	25
Wt100-3	4267	5672	33
Wt100-4	5011	6426	28
Wt100-5	5283	7709	46
Wt100-6	58258	76424	31
Wt100-7	50972	83231	63
Wt100-8	59434	90968	53
Wt50-1	2134	2832	33
Wt50-2	1996	2557	28
Wt50-3	2583	2583	0
Wt50-4	2691	3278	22
Wt50-5	1518	2568	69
Wt50-6	26276	34167	30
Wt50-7	11403	13668	20
Wt50-8	8499	9713	14
Wt40-1	913	913	0
Wt40-2	1225	1431	17
Wt40-3	537	537	0
Wt40-4	2094	2163	3
Wt40-5	990	1090	10
Wt40-6	6955	8151	17
Wt40-7	6324	9083	44
Wt40-8	6865	11474	67

Table 2: Computational results with the hybrid system.

Problem	Optimal value	Best ACS value	Best value of ACS+KA with:								
			M=1000			M=2000	M=5000			M=10000	Deviation %
			Best value	Deviation%	Computing time[s]		Best value	Deviation %	Computing time[s]		
Wt100-1	5988	8795	6310	5.38	1.906	6314	6076	1.47	8.609	6215	3.79
Wt100-2	6170	7724	6450	4.54	1.906	6182	6182	0.19	8.641	6182	0.19
Wt100-3	4267	5672	4415	3.47	1.938	4336	4372	2.46	8.688	4297	0.70
Wt100-4	5011	6426	5094	1.66	1.922	5014	5058	0.94	8.797	5069	1.16
Wt100-5	5283	7709	5433	2.84	1.938	5435	5283	0.00	8.703	5367	1.59
Wt100-6	58258	76424	60445	3.75	2.063	63804	63341	8.72	9.516	59845	2.72
Wt100-7	50972	83231	52349	2.70	2.063	55788	54822	7.55	9.484	53063	4.10
Wt100-8	59434	90968	62907	5.84	2.063	62146	62636	5.39	9.469	62817	5.69
Wt50-1	2134	2832	2134	0.00	1.063	2134	2134	0.00	4.891	2134	0.00
Wt50-2	1996	2557	1998	0.10	1.078	2009	2011	0.75	4.953	2008	0.60
Wt50-3	2583	2583	2619	1.39	1.078	2583	2583	0.00	5.031	2583	0.00
Wt50-4	2691	3278	2691	0.00	1.078	2691	2691	0.00	5.000	2691	0.00
Wt50-5	1518	2568	1518	0.00	1.078	1518	1604	5.67	5.031	1518	0.00
Wt50-6	26276	34167	27077	3.05	1.188	26758	26509	0.89	5.500	26403	0.48
Wt50-7	11403	13668	11403	0.00	1.156	11522	11733	2.89	5.375	11403	0.00
Wt50-8	8499	9713	8700	2.36	1.156	8760	8742	2.86	5.375	8700	2.36
Wt40-1	913	913	913	0.00	0.891	913	913	0.00	4.188	913	0.00
Wt40-2	1225	1324	1225	0.00	0.922	1225	1225	0.00	4.250	1225	0.00
Wt40-3	537	573	537	0.00	0.922	537	537	0.00	4.281	537	0.00
Wt40-4	2094	2098	2094	0.00	0.922	2094	2094	0.00	4.281	2094	0.00
Wt40-5	990	1090	990	0.00	0.906	990	990	0.00	4.203	990	0.00
Wt40-6	6955	12949	7024	0.99	0.984	6955	7055	1.44	4.609	6955	0.00
Wt40-7	6324	7087	6636	4.93	1.000	6324	6437	1.79	4.609	6571	3.91
Wt40-8	6865	11015	6919	0.79	1.000	6881	6919	0.79	4.594	6901	0.52

jobs at the current iteration. This list is updated dynamically for each step and each ant. Every ant keeps a copy of the best found solution until the current iteration. Each time the ant adds a new job *j* to the current sequence, this job is deleted from the copy of the best found solution. In our implementation, the first 20 jobs, which belong to the best solution and that are not already placed, form the candidate list.

Computational tests were performed in order to compare the proposed hybrid system (ACS+KA) with a stand-alone ACS. The software developed was coded in C and the tests were performed on a PC with 2330 MHz Intel processor.

The two algorithms were applied to the same instances of SMSP. Three sets of 8 problems each with 40 (Wt40-x, x=1,…,8), 50 (Wt50-x, x=1,…,8), and 100 (Wt100-x, x=1,…,8) jobs were considered. Consequently, for both, the stand-alone ACS and the hybrid system, the computational tests were done on a set of 24 instances of the SMSP. These problems were downloaded from the site http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/, which supplies data and the optimal solution for some Single Machine Scheduling Problems.

The parameters used by the ACS in the two systems are: N=10, q_0=0.9, α=0.9, β=2. The two systems evolved on the same number of iterations. Hence, the stop criterion (see fig. 2) was an upper limit for the general number of iterations. In this case this upper limit was 100.

The results obtained using only the stand-alone ACS for the 3 sets of problems are presented in table 1. The "optimal value" column contains the value of the optimization criterion for the optimal solution of the problem. For a given instance of SMSP, this value is generally unknown. The "best value" is the value of the optimization criterion for the best solution produced by the stand-alone ACS. The last column gives the deviation of the best value from the optimal one expressed in percents.

The hybrid system ACS+KA ran also over 100 general iterations, but with different values for the number of iterations, denoted M, of the stochastic searching process implemented by KA in each step 2 of the general algorithm (see fig. 2). The results are shown in Table 2.

Four values for M were considered: 1000, 2000, 5000, 10000. The deviation of the best solution from the optimal one is given only for M=5000 and M=10000 iterations. With M=10000 iterations, the hybrid system finds the optimal

solution in almost all the cases.
Despite the fact this number of iterations doesn't take much time, it is not necessary to adopt such a great number of iterations.
Let's remark we are interested in obtaining a good solution and not the optimal one, especially since it is unknown. Because the deviation is satisfactory for M=5000, this number of iterations is recommended for SMSP with 100 jobs. The same conclusion may to be drawn both for 40 or 50 jobs problems. The value of M may decrease, with very satisfactory results, to 1000 or 2000 iterations.

Comparing the two tables, one can see that the hybrid system is more efficient. Despite the fact the stand-alone ACS evolves during 100 iterations, it doesn't reach the same results as the hybrid system. When M=5000, the price to pay is the very acceptable increasing of the execution time, that means 5, 6 or 10 seconds for 40, 50 or 100 jobs problems, respectively.
The table 3 shows a comparison between the hybrid system ACS+KA and the earliest due date algorithm (EDD).

Table 3: Comparison between EDD and ACS+KA.

Problem	EDD		ACS+KA M=5000	
	Best value	Deviation %	Best value	Deviation %
Wt100-1	14138	136.11	6076	1.47
Wt100-2	19096	209.50	6182	0.19
Wt100-3	17538	311.01	4372	2.46
Wt100-4	13308	165.58	5499	9.74
Wt100-5	20218	282.70	5283	0.00
Wt100-6	13932	139.16	63341	8.72
Wt100-7	16009	214.08	54822	7.55
Wt100-8	16534	178.19	62636	5.39
Wt50-1	7306	242.36	2134	0.00
Wt50-2	7219	261.67	2011	0.75
Wt50-3	4983	92.92	2583	0.00
Wt50-4	6423	138.68	2691	0.00
Wt50-5	6257	312.19	1604	5.67
Wt50-6	57699	119.59	26509	0.89
Wt50-7	41718	265.85	11733	2.89
Wt50-8	43030	406.29	8742	2.86
Wt40-1	1588	73.93	913	0.00
Wt40-2	5226	326.61	1225	0.00
Wt40-3	3051	468.16	537	0.00
Wt40-4	5527	163.94	2094	0.00
Wt40-5	4030	307.07	990	0.00
Wt40-6	23691	240.63	7055	1.44
Wt40-7	33547	430.47	6437	1.79
Wt40-8	23032	235.50	6919	0.79

One can see that the deviation of EDD is unsatisfactory and the solution obtained with this algorithm can be only an initial solution for a more efficient algorithm, like ACS+KA.

7 CONCLUSIONS

The paper has proposed a metaheuristic for solving SMSP, implemented by a hybrid system made up of an Ant Colony System and a parallel version of the Kangaroo Algorithm.

The KA is a very simple and efficient intensifier that replaces the local optimization proposed in other papers.

The functioning of this hybrid system was compared with a stand-alone ACS. The tests have proven that this structure is more efficient than those of the simple ACS. The number of general iterations and the iterations number of the stochastic descent process are parameters of the algorithm that have to be tuned according to the size of the problem. Very good solutions were found in a quite acceptable time and number of iterations. Moreover, the increasing of the execution time is quite acceptable.

REFERENCES

Bauer A., Bullnheimer B., Hartl R.F. And Strauss C., 1999. An ant colony optimization approach for the single machine total tardiness problem, In *Proc. of CEC'99*, pages 1445–1450, IEEE Press, Piscataway, NJ,.

Beckers R., Deneubourg J.L., And Goss S., 1992. Trails and U-turns in the selection of the shortest path by the ant Lasius Niger, *Journal of Theoretical Biology*, vol. 159, pp. 397–415.

Dorigo, M., Maniezzo V. And Colorni A., 1996. The Ant System: Optimization by a colony of cooperating agents, *IEEE Transactions on Systems*, Man, and Cybernetics–Part B, 26, 29–41,

Dorigo M. And Gambardella L.M., 1997a. Ant colonies for the traveling salesman problem, *BioSystems, 43*, 73–81.

Dorigo M. And Gambardella L.M., 1997b. Ant colony system: A cooperative learning approach to the traveling salesman problem, *IEEE Transactions on Evolutionary Computation*, 1(1):53-66.

Fleury G., 1995. Application des méthodes stochastiques inspirées du recuit simulé à des problèmes d'ordonnancement, *Automatique Productique Informatique Industrielle* vol. 29-no 4-5, pp. 445-470.

Garey M; Johnson D. ,1995. Computers and Intractability: a guide to the theory on NP-completeness- New York W. H. Freeman and Co. Publishers.

Gloverf, 1989. Tabu Search- part i. ORSA *Journal of Computing* vol. 1 1989 pp. 190-206.

Kirkpatrick S. et all, 1983 -Optimization by simulated annealing- *Science* vol.220 no.4598 May 1983 pp 671-680

Madureira A. Et All, 2000. A GA Based Scheduling System for Dynamic Single Machine Scheduling Problem, ISATP 2000, Fukuoka.

Mahfoud S.; Goldberg D, 1995. Parallel recombination simulated annealing: A genetic algorithm. *Parallel computing* vol. 21 1995 pp. 1-28.

Matthijs Den Besten, Stützle T. And Dorigo M., 2000 Ant Colony Optimization for the Total Weighted Tardiness Problem, In Deb et al, editors, *Proceedings of PPSN-VI, Sixth International Conference on Parallel Problem Solving from Nature*, volume 1917 of LNCS, pages 611-620.

Minzu V. And Henrioud J.M, 1998. Stochastic algorithm for the tasks assignment in single or mixe model assembly lines, *European Journal of Automation*, Vol. 32 No 7-8, pp 831-851.

Papadimitriou C.; Steiglitz K., 1982. Combinatorial Optimization: Algorithms and Complexity - Printice-Hall.

Taillard et al., 1998. La programmation à mémoire adaptative ou l'évolution des algorithmes évolutifs. *Calculateurs parallèles* vol. 10 no. 2 April 1998 pp. 117-140.

Vaessens R. et al, 1992. A local search template. Parallel Problem Solving From Nature R. Manner and B. Manderick Edition Université Libre de Bruxelles pp. 67-76 Belgium.

SOLVING THE FEEDER ASSIGNMENT ON A REVOLVER-HEAD GANTRY MACHINE

Pyöttiälä Sami, Knuutila Timo
Department of Information Technology, University of Turku, FI-20014 Turun yliopisto, Finland
sami.pyottiala@it.utu.fi, timo.knuutila@it.utu.fi

Johnsson Mika
Valor Computerized Systems (Finland) Oy, Ruukinkatu 2, 20540 Turku, Finland
mika.johnsson@valor.com

Nevalainen Olli S.
Department of Information Technology, University of Turku, FI-20014 Turun yliopisto, Finland
olli.nevalainen@it.utu.fi

Keywords: Printed circuit boards, Electronics assembly, Multi-spindle placement machines, Revolver-head, Nozzle, Gantry machines, Optimization, Production control.

Abstract: Revolver-head gantry machines are nowadays very popular because of their flexibility, accuracy and high enough placement speed. In the optimization of this machine type the selection of nozzles into the placement head, the order of the component reels in feeder slots, and the pick-up and placement sequences have to be considered.
In this article, it is assumed that the selection of nozzles and the pick-up and placement sequences are fixed and the feeder assigment is to be solved. The problem statement is formed by analysing the operation properties of a real placement machine. Contrarily to previous literature dealing with this problem, in this work each component can be picked up only by a certain type of nozzle. Finally, four algorithms for solving the problem are proposed and tested. In the experimental tests with realistic data the algorithms performed equally.

1 INTRODUCTION

Electronic devices having at least one *printed circuit board (PCB)* inside have become more common in the last 20 years. The PCBs are used not only in the consumer electronics but also for example in cars and as parts of other bigger products. It is typical that a great number of PCBs of the same type are produced in batches and their manufacturing is done in assembly lines which normally comprise of multiple placement machines.

Research on optimisation problems of placement machines dates to late 80's (Ball and Magazine, 1988; Leipälä and Nevalainen, 1989) and it has continued actively through years to these days. There are currently several different types of placement machines in use. They all have their own technical properties and are suitable for different types of assembly tasks, see e.g. (Ayob and Kendall, 2008) for thorough review of different machines, their features and occurring problems in every machine context.

There is a lot of literature that concerns the optimisation of the operations of different types of machines except for *revolver-head gantry machines*. However, this machine type is very popular in industry at the moment. In particular, this deals with the feeder assignment and pick-up-and-place scheduling problems of this machine type. One can find two main trends in the literature on the subject. While there are methods using an evolutionary approach and solving the feeder assignment and plamecent sequence at the same time (Ho and Ji, 2004; Kulak et al., 2007), the others approach these two problems using hierarchic methods, see (Grunow et al., 2004; Lee et al., 1999; Sun et al., 2004; Ho et al., 2007).

In this article, controlling of revolver-head gantry machines is discussed. These machines are known with several names: *multi-spindle gantry placement machines*, *a revolver-head gantry machines*, *a revolver-head machines* and *collect-and-place machines*. See (Ayob and Kendall, 2008) for an illustration of the parts of this machine type. Note, that the placement head of the collect-and-place-machine can also be organised in different way so that the place-

ment head is a linear array of spindles.

In the revolver-head gantry machine the PCB to be manufactured is kept on the table at a fixed location. The placement head of the machine moves in the (x,y)-plane on the PCB and the feeder unit. The placement head is also called revolver-head or even an arm. It picks up the neccessary components from the feeder unit that is located on the side (or sides) of the machine and then mounts them on the PCB.

The placement head is moved by three step motors which run independently. The first one moves the whole gantry on its rails (in x-direction). The second motor moves the revolver-head on the gantry (in y-direction) and the third motor is used to rotate the revolver-head.

The revolver-head is equipped with multiple spindles, typically 6-12. Each spindle can hold a nozzle which can grab a component. There are many different types of component nozzles. Each component type requires a compatible nozzle. The shape of a component defines what type of nozzle should be used for grabbing it. It is also possible that a certain nozzle type is compatible with several different component types.

The feeder unit is divided into a set of feeder slots of fixed width (typically 8 mm). The slots are loaded with component tapes in which the components are stored one after the other. There are only components of the same type in each tape. Usually, the dimensions of the components are notably smaller than 8x8 mm and they fit a 8 mm tape. However, there are also wider components which are supplied in wider tapes, for example 16 mm. These tapes occupy more than one feeder slots but their use is similar to that of narrow tapes.

At a high abstraction level, revolver-head gantry machines operate in cycles of four phases:

1. *pick-up*-phase
2. *travel onto the PCB* -phase
3. *placement*-phase
4. *travel onto the feeder* -phase

While there are slight differences in the operation principles of different machines, we consider a case, where the design of the machine includes the following details: In the first phase, the revolver-head moves on the feeder unit and collects one component after the other from the right feeder slots keeping the component tapes. Between two pick-up events the revolver-head has to rotate at least one step to get an empty nozzle operable. It is also possible that the revolver-head has to move in the x-direction (which is parallel to the length of the feeder unit) onto a correct feeder slot. The revolver can rotate while it is moving to the next pick-up location. In the pick-up-phase, we suppose that the revolver never rotates more than 360° and it can leave one or more nozzles empty by skipping them during the pick-ups.

After at least one component has been picked up the placement head moves onto the PCB area. This is called travel onto the PCB -phase. During this movement, the revolver rotates so that the component which was picked up first in the pick-up-phase can be placed immediately after the head has reached the placement location.

In the placement-phase, the collected components are placed onto the PCB in their correct locations in the same order as they were picked up. The rotation limit of 360° concerns this phase, also. Finally, the empty revolver-head is moved back onto the feeder unit and rotated so that the suitable nozzle for the next component to be picked up is operable again.

If for example 180 components have to be placed on a single PCB the gantry machine with revolver spindle count of 12 makes at least 15 tours described above. However, this requires that all the nozzles are loaded in every single tour. In practice, it is usually impossible to get full loads because of component-nozzle incompatibilities. If the next nozzle in the revolver is not compatible with the next component that should be picked up, the revolver is rotated more than one step forward.

There are placement machines that can change nozzles automatically during the manufacturing process but in many cases the change-process takes too much time and it is therefore avoided. On the otherhand, in some placement machines the nozzle changes can be done only manually and the placement process has to be stopped for doing that.

A setup operation (i.e. change of component tape reels, conveyor belt adjustment etc.) of the placement machine must be performed before the machine can manufacture any new PCB types. At least the following decicions have then to be made:

1. assign the component types into the slots of the feeder unit,
2. define the set of component nozzles and their order in the revolver-head, and
3. define the sequence and tours in which components are picked up and placed.

It has been commonly assumed in previous literature that each nozzle can grab any type of component exist, so there has not been a need to solve the nozzle selection for the revolver-head. However, the situation is often more complicated in practice. The consideration of the nozzle-component compatibility

is one of the aspects which tends to make the design of the machine control more difficult. If it is assumed (as it is often done) that every nozzle can pick up any type of component in a single machine then we also have to assume that there are (even for narrow component types) multiple machines in the production line and components are divided among them so that universal nozzle compatibility is achieved in every machine. Algorithms for deciding the nozzle selection of the revolver are proposed in (Pyöttiälä et al., 2006).

The goal of this article is to find such a feeder setup that minimizes *the assembly cycle time (ACT)* of the single multi-spindle gantry machine. We suppose that sequence of component placements and the assignment of nozzles for the head are fixed. We thus consider a subproblem of the total machine control problem. It is assumed that a significant number of PCBs of the same type are manufactured. This set of PCBs is called a *batch*. Here, *ACT* is the time required to manufacture a single PCB in the batch. Clearly, minimizing *ACT* minimizes also the time requirement of the whole batch.

We formulate the *Optimal Feeder Assignment problem (OFA)* in Section 2. The formulation is then used in four heuristics to solve the problem in Section 3. In this formulation we omit the possibility of duplicate compoment tapes in the feeder unit and the delays caused by camera inspections. Further, we suppose that the nozzle-to-arm assignment is fixed for the whole PCB job and revolver rotations can be done in parallel with the head movements. Results of the heuristics are compared with a lower bound in Section 4.

2 PROBLEM STATEMENT

2.1 Notation and Terminology

The discussion of the previous section leads to the following notation which can be used to describe the determination of *ACT*.

- the set of component types $CT = \{ct_1, \ldots, ct_m\}$
- the recipe of a PCB *i.e.* the set of components and their locations $C = \{c_1, \ldots, c_n\}$, where $c_i = (t, (x_i, y_i))$ such that $t \in CT$ and $(x_i, y_i) \in \mathbb{R}^2$
- the set of nozzle types $NT = \{nt_1, \ldots, nt_l\}$
- for convenience, the location of a component on the PCB is also given by function $cl : C \mapsto \mathbb{R}^2$
- the type of component is given by function $ct : C \mapsto CT$
- the location of the component of a certain type in the feeder is given by function $fl : CT \mapsto \mathbb{R}^2$
- the nozzle requirement of a certain component type is given by function $nt : CT \mapsto NT$
- the nozzle sequence of the revolver-head *i.e.* an *arm* is $\alpha = (n_1, \ldots, n_k)$, where $n_i \in NT$
- distance (Chebychev) between two locations is given by function $d : \mathbb{R}^2 \times \mathbb{R}^2 \mapsto \mathbb{R}$ so that $d((x_1, y_1), (x_2, y_2)) = max(|x_1 - x_2|, |y_1 - y_2|)$
- time required by the revolver-head to *travel* a certain distance is given by function $tt : \mathbb{R} \mapsto \mathbb{R}$
- function $re : C \times \alpha \mapsto \mathbb{N}$ gives the number of rotation steps required to rotate the revolver-head so that the next suitable empty nozzle can pick up a certain component
- function $rf : C \times \alpha \mapsto \mathbb{N}$ gives the number of rotation steps required to rotate the revolver-head so that the next nozzle that holds a component can place it
- time required by the revolver-head to *rotate* certain steps forward is given by function $rt : \mathbb{N} \mapsto \mathbb{R}$
- pt is a constant time that a single pick-up or placement takes

A permutation of PCB recipe C is called a *job* and it determines an order in which the components can be placed on the PCB. Each component of set C occurs exactly once in a permutation. Often, we use symbol W for a job and it can be partitioned into p separate subjobs such that $W = W_1 \cdot W_2 \cdots W_p$. (The partitioning is discussed for example in (Knuutila et al., 2007).)

Definition 1.1 Given arm α of size k and any partition of job $W_i = (c_1^i, c_2^i, \ldots, c_s^i)$, where $s \leq k$. We say that α *can pick up* W_i if and only if W_i is a *subsequence* of α.

Definition 1.2 Given arm α and job $W = W_1 \cdot W_2 \cdots W_p$. We say that α *can execute* W if and only if α can pick up each W_i.

2.2 Assembly Cycle Time

Suppose that job W of length n, arm α of size k and functions as above in section 2.1 are given. Further, let us assume that α can execute W in p separate pick-up and placement tours and the revolver-head (arm) is initially located at the park position (x_{pp}, y_{pp}). The assembly cycle time can then be defined as follows

$$ACT(W, \alpha) = cost_1 + \sum_{i=2}^{p-1} \left(cost_i \right) + cost_p,$$

where $cost_1$ is the time consumed in travelling from park position to the location of the feeder slot of the first component type, $cost_i$ is the time that passes in the ith tour of the process and $cost_p$ is the time of the last tour and travel time from last placement location to the park position. We denote by

$$cost_1 = tt(d((x_{pp}, y_{pp}), fl(ct(c_1^1)))),$$

$$cost_i = pt + \sum_{j=1}^{|W_i|-1} (cost_{i,j}^{pickup}) + cost_i^{travel\_PCB} + pt + \sum_{j=1}^{|W_i|-1} (cost_{i,j}^{place}) + cost_i^{travel\_feeder},$$

and

$$cost_p = pt + \sum_{j=1}^{|W_p|-1} (cost_{p,j}^{pickup}) + cost_p^{travel\_PCB} + pt + \sum_{j=1}^{|W_p|-1} (cost_{p,j}^{place}) + tt(d(cl(c_{|W_p|}^p), (x_{pp}, y_{pp}))),$$

furthermore,

$$cost_{i,j}^{pickup} = max\left(tt\left(d\left(fl(ct(c_j^i)), fl(ct(c_{j+1}^i))\right)\right), rt\left(re(c_{j+1}^i, \alpha)\right)\right) + pt,$$

$$cost_i^{travel\_PCB} = tt(d(fl(ct(c_{|W_i|}^i)), cl(c_1^i)),$$

$$cost_{i,j}^{place} = max\left(tt\left(d\left(cl(c_j^i), cl(c_{j+1}^i)\right)\right), rt\left(rf(c_{j+1}^i, \alpha)\right)\right) + pt,$$

and

$$cost_i^{travel\_feeder} = tt(d(cl(c_{|W_i|}^i), fl(ct(c_1^{i+1})).$$

Note, that revolver-head rotations and (x,y)-travels are simultaneous operations and one has therefore considered which one of these takes longer time (c.f. formulae for $cost_{i,j}^{pickup}$ and $cost_{i,j}^{place}$).

We still summarize the assumptions relating the operation principle of the placement machine.

- Placement head may rotate at most 360° in a single pick-up phase. This concerns also the placement phase.
- The nozzle setup of arm α has at least one nozzle of each type that the executing of job W requires.
- There can be multiple nozzles of the same type in the arm.
- Note, that in our model the number of nozzle types required by W is ≥ 1 instead of being $= 1$. (This makes a difference from earlier literature on revolver-head gantry machines.)

Now, the main problem of this work can be stated as

Optimal Feeder Assignment Problem (OFA). Given recipe C, job W and its partition $W = W_1 \cdot W_2 \cdots W_p$, arm α of size k and functions as above (except fl). Assign component types to feeder slots so that $ACT(W, \alpha)$ is minimized.

3 SOLVING OFA

In this section four different strategies to solve the OFA problem are proposed. The first strategy evaluates simply a set of random feeder assignments. In the second strategy, the neighborhood of the components on the PCB is analysed and the feeder assignment rests on that analysis. This method has been proposed before in (Grunow et al., 2004). The last two strategies are based on the frequencies of the different component types in a PCB recipe. The four methods are described in more detail in the following subsections.

3.1 Random Sample Feeder Assignment

The random strategy (random) to decide the feeder assignment is straightforward: a constant number (in our case 10000) of suitable random feeder assignments are formed and their *ACT*s are calculated. The feeder assignment with the lowest *ACT* is the result of the search process.

3.2 MST-based Feeder Assignment

Grunow et al. (Grunow et al., 2004) used a minimum spanning tree (MST) for the analysis of the neighbourhood of components on the PCB. Here, the same idea is applied in algorithm mst.

At the beginning, a fully connected graph of all component placement points on the PCB is formed and the MST of this graph is solved. In the next phase the number of the neighbours in the MST is calculated for each component type. Then the different component types are put in a priority queue using the number of the neighbours as a priority index. Finally, the

component type with the highest priority index gets the feeder slot which is at the center of the feeder unit. The component type with the second highest index is then put on the left side of the first one and the third goes to the right side of the first. This dealing process is iterated until all component types in the queue have been assigned to some feeder slot.

3.3 Frequency-based Feeder Assignment

In this method (freq), the frequencies of the component types of the PCB recipe are calculated and used as priority indices. The feeder slots are then occupied in the same way as in 3.2.

3.4 Frequency-based Balanced Feeder Assignment

Frequency based feeder assignment of subsection 3.3 may lead into a very unbalanced configuration if there are big differences in the number of every second component type. For example, the left half of the feeder unit may feed a significant amount of the components. This can be avoided if the next component type in the priority queue is always assigned on the side which has less components to feed at the moment. This heuristic (freq_bal) implements the balancing property which leads to a more balanced outcome in terms of feeded components. However, the other parts of this method follow the heuristic of subsection 3.3 closely.

4 EXPERIMENTAL TESTS

In this section, the results of a set of numerical test with the four heuristics of section 3 are discussed. There are 43 different PCB recipes based on genuine products but they differ from originals slightly. All of them are parts of the actual PCBs because the original PCBs were assembled in a production line of four different placement machines and this article concerns the production control of a single machine. The modifications of the PCBs were done by deleting random periods from the original recipies so that the requirements of different nozzles and the number of feeder slots needed met the limits of a single gantry machine.

The characteristics of the test PCB-recipe set are shown in Table 1. The technical properties of the placement machine are presented in Table 2 and they mainly follow those used in (Kulak et al., 2007).

Table 1: Dimensions and number of components and component types in the tests with 43 PCBs.

min number of comp. on PCB	34
max number of comp. on PCB	199
smallest board size (mm)	125.5 x 127.8
largest board size (mm)	400 x 255.2
min number of comp. types on PCB	1
max number of comp. types on PCB	40

During the tests, the nozzle selection of the revolver-head and the placement sequence were fixed. The feeder assignment was then solved using the heuristics and *ACT* was calculated for each of the solutions. *A theoretic lower bound* (described below) was also calculated using the written software.

The average values of *ACT*s of the 43 test cases are shown for the heuristics in Table 3. The table also shows an average *ACT* for the theoretic lower bound (theoretic) in which the *ACT* has been calculated for the case where all components are picked up from the center of the feeder unit and placed onto a placement point which is closest to the center slot of the feeder. The *ACT*s for all 43 test cases are presented graphically in Figure 1. The figure shows that there is no statistically significant difference (tested in Excel, t-test) in the efficiency of the different heuristics.

However, the *ACT*s calculated from the results of heuristics were clearly longer than the theoretic lower bound yielded which was naturally expected.

Because the test results of the different methods are equal it can be asked if the optimization of the placement process of the revolver-head gantry machines should be focused more on the placement sequencing and on selecting a suitable combination of nozzles into the revolver-head than on the feeder assignment.

Heuristics proposed in this article give a feasible enough solution for the feeder assignment problem and the placement sequence can be solved for a feeder assignment which has been determined by OFA. (Of course, it is also possible to solve these problems at the same time, for example using genetic approach, see (Kulak et al., 2007).) Even the trivial sampling method (in section 3.1) that randomly generates different solutions produced as good results as the more advanced heuristics. This encourages to think that because the rotation of the revolver takes time anyway between two pick-ups, the order of the component types in the feeder is not so critical when it comes to *ACT*.

Table 2: Technical properties of the placement machine in simulations.

velocity in x- and y-axis	800 mm/s
step rotate time	0.05 s
single component placement time	0.04 s
single component pick-up time	0.04 s
feeder slot sizes	8.0, 12.0, 16.0 mm
number of spindles	12
feeder slot capacity	40

Table 3: Average of *ACT*s for four heuristics and the value of the theoretic lower bound.

theoretic	34.85
random	41.29
mst	41.46
freq	41.50
freq_bal	41.43

5 CONCLUSIONS

In this article, the control of a revolver-head gantry machine was discussed and its properties were described. A mathematical model for the assembly cycle time was given for the machine type and the optimal feeder assignment problem (OFA) was formulated.

Four heuristics were proposed for solving OFA of the revolver-head gantry machine. The heuristics were tested and the test results were compared to each other and a the theoretic lower bound. The heuristics performed equally when tests were done with realistic PCB assembly data. Since the different methods yield different solutions to OFA and it still does not have a significant affect on *ACT*, the other problems in the placement machine control context may be more important objects for optimization. For example, selecting a good nozzle combination into the revolver-head and pick-up- and place-sequencing seems to be such a problem.

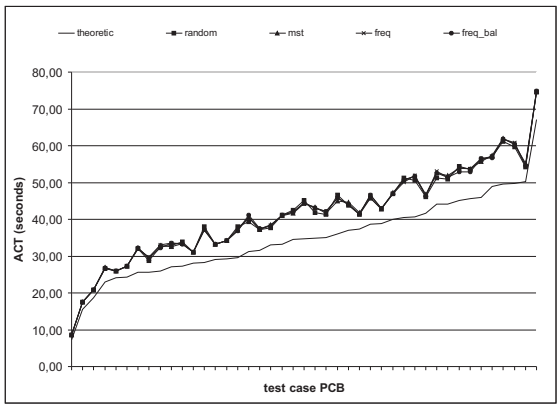


Figure 1: Assembly cycle times of heuristics and theoretic lower bound for all 43 test PCB-recipies.

REFERENCES

Ayob, M. and Kendall, G. (2008). A survey of surface mount device placement machine optimisation: Machine classification. In *European Journal of the Operational Research, vol. 186, pp. 893-914*.

Ball, M. and Magazine, M. (1988). Sequencing of insertions in printed circuit board assembly. In *Operations Research 36(2), pp. 192-201*.

Grunow, M., Günther, H.-O., Schleusener, M., and Yilmaz, I. (2004). Operations planning for collect-and-place machines in pcb assembly. In *Computers & Industrial Engineering, Vol. 47, pp.409-429*.

Ho, W. and Ji, P. (2004). A hybrid genetic algorithm for component sequencing and feeder arrangement. In *Journal of Intelligent Manufacturing, 15, pp. 307-315*.

Ho, W., Ji, P., and Wu, Y. (2007). A heuristic approach for component scheduling on a high-speed pcb assembly machine. In *Production Planning & Control, Vol. 18, No. 8, pp. 655-665*.

Knuutila, T., Pyöttiälä, S., and Nevalainen, O. (2007). Minimizing the arm movements of a multi-head gantry machine. In *Proceedings of 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007)*.

Kulak, O., Yilmaz, I., and Günther, H.-O. (2007). Pcb assembly scheduling for collect-and-place machines using genetic algorithms. In *International Journal of Production Research, Vol. 45, pp. 3949-3969 No. 17*.

Lee, S., Lee, H., and Park, T. (1999). A hierarchical method to improve the productivity of a multi-head surface mounting machine. In *Proc. of the IEEE International conference on Robotics and Automation*.

Leipälä, T. and Nevalainen, O. (1989). Optimization of the movements of a component placement machine. In *European Journal of Operational Research, 38, pp. 167-177*.

Pyöttiälä, S., Knuutila, T., and Nevalainen, O. (2006). The selection of nozzles for minimizing the number of pick-ups on a multi-head placement machine. In *GTCM2006 Conference, Groningen, The Netherlands*.

Sun, D.-S., Lee, T.-E., and Kim, K.-H. (2004). Component allocation and feeder arrangement for a dual-gantry multi-head surface mounting placement tool. In *International Journal of Production Economics 95 pp. 245-264*.

PERFORMANCES IMPROVEMENT AND STABILITY ANALYSIS OF MULTIMODEL LQ CONTROLLED VARIABLE-SPEED WIND TURBINES

Nadhira Khezami[1,2], Xavier Guillaud[2] and Naceur Benhadj Braiek[1]
[1]*LECAP, École Polytechnique de Tunisie, BP 743 – 2078 La Marsa, Tunisia*
[2]*L2EP, École Centrale de Lille, Cité Scientifique, 59651 Villeneuve d'Ascq Cedex, France*
nadhira.khezami@ec-lille.fr, xavier.guillaud@ec-lille.fr, naceur.benhadj@ept.rnu.tn

Keywords: LQ controller, Multimodel approach, Global stability, LMI, Lyapunov equations.

Abstract: In this paper, a linear quadratic (LQ) control law combined with a multimodel approach is designed for variable-speed, variable-pitch wind turbines. The presented technique is based on an optimal control method in order to improve the system global dynamic. A set of linear local models (sub-models) is then defined for different operating points corresponding to high wind speeds. Thereafter, a global asymptotic stability analysis is developed by solving a bilinear matrix inequality (BMI) feasibility problem based on the local stability of the sub-models.

1 INTRODUCTION

Nowadays, the growth of the utilization of the wind turbines is more and more important since they are producing carbon-emission-free electricity. Until today, only classic control laws, such as P, PI or PID controllers, are used in the wind turbines. However, the performance of these controllers is limited by the high nonlinear characteristics of the wind turbine and by the appearance of new control objectives required by the grid-codes; the reason why advanced control research area is improving every day.

In the first axis of this paper, an LQ controller, which had been advocated by many researchers, is designed with a multimodel approach, for pitch regulated variable speed wind turbines operating at high wind speeds, in order to guarantee an optimal behavior for the studied process. However, this technique still presents some limits to satisfy all the control objectives especially those concerning the system global dynamic. This paper aims then to present an issue for this problem by adding an exponential term in the quadratic cost function.

The second section deals with the asymptotic stability analysis of the global system by solving a set of BMI according to the Lyapunov theorems. In fact, the stability study is necessary and important to illustrate the effectiveness of the presented strategy.

Finally, the simulation results realized on Matlab Simulink are presented and discussed.

2 WIND TURBINE MODELLING

2.1 Wind Turbine Description

The considered wind turbine (Figure 1) is modeled as two inertias (the generator and the turbine inertias respectively J_g and J_T) linked to a flexible shaft with a mechanical coupling damping coefficient d and a mechanical coupling stiffness coefficient k. This model is widely used in the literature (Bianchi *et al.*, 2004; Camblong *et al.*, 2002).

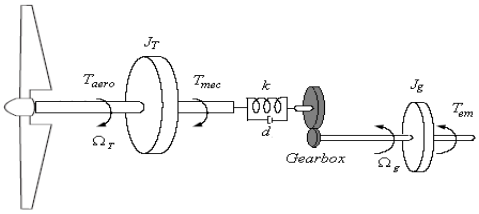


Figure 1: Wind turbine dynamic model.

where Ω_T and Ω_g are the turbine and the generator rotational speeds, T_{em} is the generator torque, T_{mec} is the drive train mechanical torque and T_{aero} is the torque caught by the wind turbine which is expressed by:

$$T_{aero} = \tfrac{1}{2} \cdot \frac{\rho \cdot \pi \cdot R^5 \cdot \Omega_T^2}{\lambda^3} . c_p(\lambda, \beta) \qquad (1)$$

where ρ is the air density and R is the turbine radius.

The power coefficient c_p (Figure 2) is a non linear function of the blade pitch β and the tip speed ratio λ depending on the wind speed value v and given by:

$$\lambda = \frac{\Omega_T \cdot R}{v} \tag{2}$$

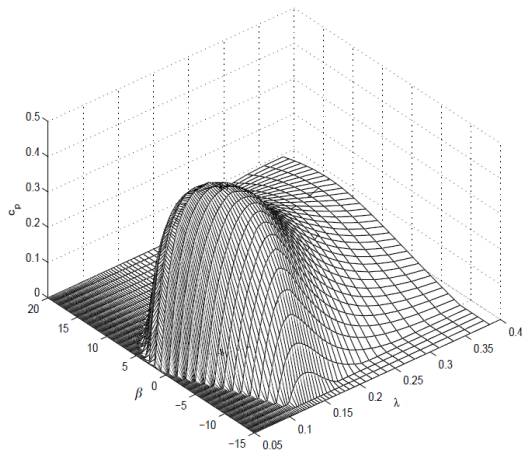


Figure 2: Power coefficient curves.

The dynamic response of the rotor is given by:

$$J_T.\dot{\Omega}_T = T_{aero} - T_{mec} \tag{3}$$

The generator is driven by the mechanical torque and braked by the electromagnetic torque. Reported to the low speed shaft, the characteristic equation is the following:

$$J_{g-ls}.\dot{\Omega}_{g-ls} = T_{mec} - G.T_{em} \tag{4}$$

where G is the gearbox gain and:

$$\begin{aligned} - \quad & \Omega_{g-ls} = \frac{\Omega_g}{G} \\ - \quad & J_{g-ls} = G^2.J_g \end{aligned} \tag{5}$$

And the low speed shaft torque T_{mec} results from the torsion and friction effects due to the difference between the generator and the rotor speeds (Boukhezzara *et al.*, 2007). It's defined by the following equation reported to the low speed shaft:

$$\dot{T}_{mec} = k.(\Omega_T - \Omega_{g-ls}) + d.(\dot{\Omega}_T - \dot{\Omega}_{g-ls}) \tag{6}$$

The pitch actuator dynamic is described by a first order system:

$$\dot{\beta} = \frac{1}{\tau_\beta}(\beta_{ref} - \beta) \tag{7}$$

β_{ref} represents the control value of the blade-pitch angle β and τ_β is the time constant of the pitch actuator.

2.2 Linearization and State Representation

The wind turbine is a complex non linear system presenting several difficulties in study and control. It seems then more suitable to describe it with a set of linear local models valid in different operating points corresponding to different levels of wind speed values. The principle of this method is used in several techniques. In this paper, we use the multimodel approach which was the subject of many research works (Kardous *et al.*, 2006, 2007).

For the studied system, we define a multimodel base made of four local models. The equivalent instantaneous model, as described in Figure 3, is obtained by a fusion of only two valid successive models. The choice of these models depends on the wind speed value.

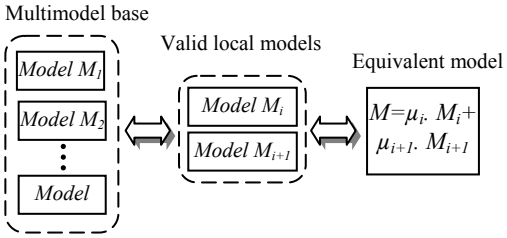


Figure 3: Wind-turbine multimodel description.

The weighting coefficient μ_i is the validity value of the model M_i and it can be expressed by:

$$\mu_i = 1 - r_i \tag{8}$$

r_i is a normalized residue measuring the error between the instantaneous and the valid local model wind speed values (respectively v and v_i). When M_i and M_{i+1} are the valid models, the residue can be expressed as:

$$r_i = \frac{|v_i - v|}{v_i + v_{i+1}} \tag{9}$$

Thus, the validities satisfy the convex sum, such that: $\mu_i + \mu_{i+1} = 1$

To obtain the local models, the system should be then linearized around the operating point. The non-linearity of the system is due to the c_p characteristic which is used in the expression of the aerodynamic torque. We need then to linearize the expression (1)

of T_{aero} around an operating point ($o.p$) defined by the wind speed value v_i (Bianchi *et al.*, 2007; Munteanu *et al.*, 2005). We can define:

$$\begin{aligned}\Delta T_{aero} &= \left.\frac{\partial T_{aero}}{\partial \Omega_T}\right|_{o.p} .\Delta\Omega_T + \left.\frac{\partial T_{aero}}{\partial \beta}\right|_{o.p} .\Delta\beta \\ &= a_i.\Delta\Omega_T + b_i.\Delta\beta\end{aligned} \quad (10)$$

where:

$$\begin{cases} a_i = \frac{1}{2}.\rho.\pi.R^3.\frac{v_i^2}{\Omega_{T\_nom}}.\left[\frac{\partial c_p(\lambda,\beta)}{\partial\lambda} - \frac{c_{p_{i\_nom}}}{\lambda_{i\_nom}}\right] \\ b_i = \frac{1}{2}.\rho.\pi.R^2.\frac{v_i^3}{\Omega_{T\_nom}}.\frac{\partial c_p(\lambda,\beta)}{\partial\beta} \end{cases} \quad (11)$$

and:

$$\begin{aligned} c_{p_{i\_nom}} &= c_p\left(\lambda_{i\_nom}, \beta_{i\_nom}\right) \\ &= \frac{2.\Omega_{T\_nom}.T_{aero\_nom}}{\rho.\pi.R^2.v_i^3} \end{aligned} \quad (12)$$

The symbol Δ represents the deviation from the chosen operating point corresponding to (Ω_{T_nom}, $\Omega_{g\text{-}ls\text{-}nom}$, $\beta_{i\text{-}nom}$, $T_{mec\text{-}nom}$, $T_{em\text{-}nom}$ and P_{nom}) where: $T_{em\text{-}nom}$ and $T_{mec\text{-}nom}$ are respectively the nominal values of the electromagnetic and the mechanical torques.

Thereafter, the linearization of the non-linear system expressed in equations (3), (4), (6) and (7) around an operating point gives a state space representation of the form below:

$$\begin{cases} \dot{x} = A_i.x + B_i.u \\ y = C_i.x + D_i.u \end{cases} \quad (13)$$

where x, u and y are respectively the state, control and output vectors defined as:

$$x = \begin{bmatrix} \Delta\Omega_T \\ \Delta\Omega_{g-ls} \\ \Delta\beta \\ \Delta T_{mec} \end{bmatrix}, y = \begin{bmatrix} \Delta\Omega_T \\ \Delta P \end{bmatrix} \text{ and } u = \begin{bmatrix} \Delta\beta_{ref} \\ \Delta T_{em} \end{bmatrix} \quad (14)$$

Notice that $P=T_{em}.\Omega_g$ designates the generated electrical power. This leads to write around an operating point:

$$\Delta P = G.\left(T_{em\_nom}.\Delta\Omega_{g-ls} + \Delta T_{em}.\Omega_{g-ls\_nom}\right) \quad (15)$$

Hence, A_i, B_i, C_i and D_i, which are respectively the state, input, output and feedthrough matrices, are defined as follows:

$$\begin{cases} A_i = \begin{pmatrix} \frac{a_i}{J_T} & 0 & \frac{b_i}{J_T} & -\frac{1}{J_T} \\ 0 & 0 & 0 & \frac{1}{J_{g-ls}} \\ 0 & 0 & -\frac{1}{\tau_\beta} & 0 \\ k+\frac{a_i.d}{J_T} & -k & \frac{d.b_i}{J_T} & -d.\left(\frac{1}{J_T}+\frac{1}{J_{g-ls}}\right) \end{pmatrix} \\ B_i = \begin{pmatrix} 0 & 0 \\ 0 & -\frac{G}{J_{g-ls}} \\ \frac{1}{\tau_\beta} & 0 \\ 0 & \frac{d.G}{J_{g-ls}} \end{pmatrix}, C_i = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & G.T_{em\_nom} & 0 & 0 \end{pmatrix} \\ \text{and } D_i = \begin{pmatrix} 0 & 0 \\ 0 & G.\Omega_{g-ls\_nom} \end{pmatrix} \end{cases} \quad (16)$$

3 CONTROLLER DESIGN

The control task is based on the objective of regulating the rotor rotational speed and the generated power by acting on two control variables: the electromagnetic torque T_{em} and the regulating pitch angle β_{ref}.

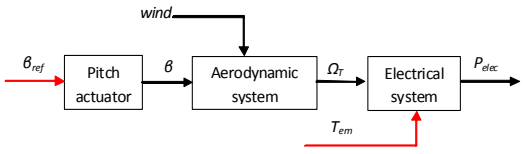


Figure 4: Wind turbine block diagram.

The LQ control strategy had been advocated by many research works (Boukhezzara *et al.*, 2007; Khezami *et al.*, 2009; Poulsen *et al.*, 2005; Hammerum et al., 2007; Cutululis *et al.*, 2006). This technique presents a good compromise between the performances optimization and the minimization of the control signals by the use of a quadratic cost function. However, it also presents the disadvantage of the non possibility of controlling the global system dynamic. In this paper, a solution that can partially solve this problem is presented.

This controller aims to minimize the following quadratic criterion J:

$$J = \frac{1}{2}\int_0^{+\infty}\left(y^T.Q.y + u^T.R.u\right)e^{2\alpha t}dt \quad (17)$$

where Q and R are diagonal positive definite matrices.

The term $y^T.Q.y$ expresses the performances optimization, the term $u^T.R.u$ expresses the minimization of the control signals and the term $e^{2\alpha t}$ allows the performances improvement of the classic quadratic criterion. It leads to the placement of the system poles on the left of $-\alpha$.

The criterion can be rewritten as follows with an input-state cross term:

$$J = \frac{1}{2}\int_0^{+\infty} \left(x^T.Q_1.x + 2.x^T.N.u + u^T.R_1.u \right) e^{2\alpha t} dt \quad (18)$$

where Q_1, R_1 and N are defined as :

$$\begin{cases} Q_1 = C^T.Q.C \\ R_1 = R + D^T.Q.D \\ N = C^T.Q.D \end{cases} \quad (19)$$

For this criterion, the optimal gain can be calculated from the following Riccati equation:

$$\begin{cases} {A_{i\alpha}}^T.L + L.A_{i\alpha} - (L.B_i + N_1).R_1^{-1}.(B_i^T.L + N_1^T) \\ \quad + Q_1 = 0 \\ A_{i\alpha} = A_i + \alpha.I \\ K_i = R_1^{-1}.(B_i^T.L + N_1^T) \end{cases} \quad (20)$$

where I is the identity matrix.

Since the dynamic of the pitch actuator should not be changed, the controller is designed in two steps. In the first step, we consider the blade pitch angle β and the electromagnetic torque T_{em} as control variables instead of β_{ref} and T_{em}. The state representation becomes then:

$$\begin{cases} \dot{x}_1 = A_{i1}.x_1 + B_{i1}.u_1 \\ y = C_{i1}.x_1 + D_{i1}.u_1 \end{cases} \quad (21)$$

where:

$$\begin{cases} x_1 = \begin{bmatrix} \Delta\Omega_T \\ \Delta\Omega_{g-ls} \\ \Delta T_{mec} \end{bmatrix}, y = \begin{bmatrix} \Delta\Omega_T \\ \Delta P \end{bmatrix} \text{ and } u_1 = \begin{bmatrix} \Delta\beta \\ \Delta T_{em} \end{bmatrix} \\ A_{i1} = \begin{pmatrix} \frac{a_i}{J_T} & 0 & -\frac{1}{J_T} \\ 0 & 0 & \frac{1}{J_{g-ls}} \\ k + \frac{a_i.d}{J_T} & -k & -d.\left(\frac{1}{J_T} + \frac{1}{J_{g-ls}} \right) \end{pmatrix} \\ B_{i1} = \begin{pmatrix} \frac{b_i}{J_T} & 0 \\ 0 & -\frac{G}{J_{g-ls}} \\ \frac{d.b_i}{J_T} & \frac{d.G}{J_{g-ls}} \end{pmatrix}, \\ C_{i1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & G.T_{em-nom} & 0 \end{pmatrix} \\ \text{and } D_{i1} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \end{cases} \quad (22)$$

From this representation, the optimal gain $K_{i1} = \begin{bmatrix} K_{i1\_\beta} \\ K_{i1\_Tem} \end{bmatrix}$ is calculated such that:

$$u_1 = \begin{bmatrix} \Delta\beta \\ \Delta T_{em} \end{bmatrix} = -K_{i1}.x_1 \quad (23)$$

The relation (23) leads to the following optimal control law using the global state vector as shown in Figure 5:

$$\begin{cases} u = -K_i.x \\ K_i = \begin{bmatrix} K_{i\_\beta ref} \\ K_{i\_Tem} \end{bmatrix} \end{cases} \quad (24)$$

with:

$$\begin{cases} K_{i\_\beta ref} = \left(K_{i1\_\beta} + \tau_\beta.K_{i1\_\beta}.A_{i1} \right).T_1 + \tau_\beta.K_{i1\_\beta}.B_{i1\_\beta}.T_2 - \\ \qquad \tau_\beta.K_{i1\_\beta}.B_{i1\_Tem}.K_{i1\_Tem}.T_1 \\ K_{i\_Tem} = K_{i1\_Tem}.T_1 \end{cases} \quad (25)$$

and:

$$\begin{cases} \begin{bmatrix} B_{i1\_\beta} & B_{i1\_Tem} \end{bmatrix} = B_{i1} \\ T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ T_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \end{cases} \quad (26)$$

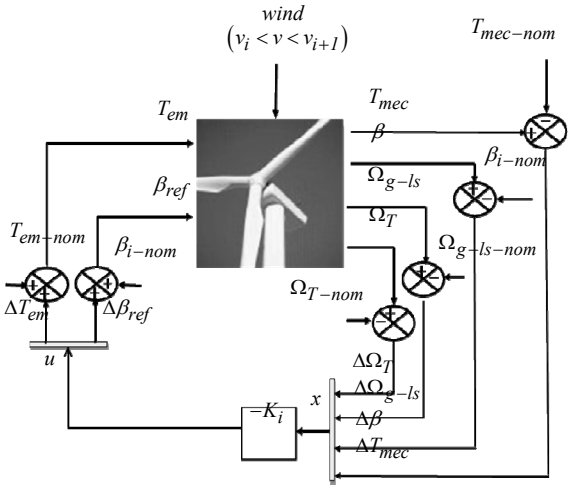


Figure 5: The LQ controller design.

4 STABILITY STUDY

The quantum advance in stability theory that allowed one the analysis of arbitrary differential equations is due to Lyapunov, who introduced the

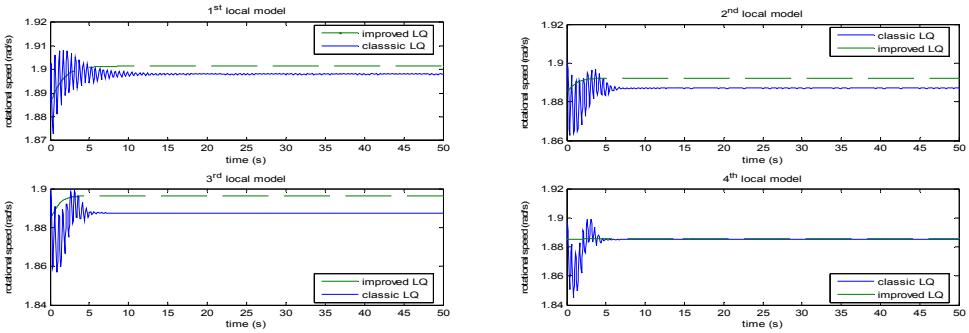


Figure 6: Comparison simulation between classic and improved LQ control laws.

basic idea and the definitions of stability that are in use today. The concept of Lyapunov stability plays an important role in control and system theory.

As we define a global model M by fusion of two successive local models M_i and M_{i+1}, the characteristic matrices of the system (13) can be obtained by:

$$\begin{cases} A_i = \mu_k.A_k + \mu_{k+1}.A_{k+1} \\ B_i = \mu_k.B_k + \mu_{k+1}.B_{k+1} \\ C_i = \mu_k.C_k + \mu_{k+1}.C_{k+1} \\ D_i = \mu_k.D_k + \mu_{k+1}.D_{k+1} \end{cases} \quad (27)$$

The input vector is calculated by:

$$u = -\left(\mu_k.K_k + \mu_{k+1}.K_{k+1}\right).x \quad (28)$$

Hereafter, the state vector can be represented as:

$$\begin{aligned} \dot{x} &= \sum_i \sum_j \mu_i.\mu_j.\left(A_i - B_i.K_j\right).x \\ &= \sum_i \mu_i^2.G_{ii}.x + 2.\sum_{j=i+1} \mu_i.\mu_j.\left(\frac{G_{ij}+G_{ji}}{2}\right).x \end{aligned} \quad (29)$$

where: $G_{ij} = A_i - B_i.K_j$

To study the global asymptotic stability of the above system supplied by the multimodel LQ control, the first necessity is to analyze the stability of every local model. As we focus here especially on the closed-loop system, the criterion of stabilization consists then in finding, for a local model M_i, a positive definite matrix P that satisfies the following LMI (Chedli, 2002; Liberzon and Morse, 1999):

$$G_{ii}^T.P + P.G_{ii} < 0 \quad (30)$$

In the case of the multimodel systems, an extra condition is to add to the LMI (30) in order to guarantee the global stability (Chedli, 2002; Liberzon and Morse, 1999; Kardous *et al.*, 2003) and it consists in:

$$Q_{ij}^T.P + P.Q_{ij} < 0, \quad i < j \quad (31)$$

where $Q_{ij} = \dfrac{G_{ij} + G_{ji}}{2}$

And in our case, only two successive local models are valid at a time, which means that this condition will be considered for *i=1 to 3* and *j=i+1*.

5 SIMULTATION RESULTS

The proposed control approach and the stability analysis of the controlled system have been illustrated through simulations on Matlab Simulink.

The simulated wind turbine parameters are presented in Table 1.

To calculate the linearization coefficients a_i and b_i, the following c_p empiric expression relative to 2MW wind turbines is used:

$$c_p = 0.18\times\left(\frac{90}{0.4+0.5\lambda} - 6.8 - 0.115\beta^2\right)\times e^{\frac{-8}{0.4+0.5\lambda}+0.16} \quad (32)$$

Table 1: Wind turbine parameter values.

Parameters		*Values*
Air density	ρ	*1,22 Kg/m³*
Turbine radius	R	*40m*
Nominal power	P_{nom}	*2MW*
Nominal speed	$\Omega_{T\text{-}nom}$	*18 rpm*
Optimal power coefficient	$c_{p\text{-}opt}$	*0.4775*
Optimal speed ratio	λ_{opt}	*9*
Gearbox gain	G	*92.6*
Turbine inertia	JT	4.9×10^6 *N.m.s²*
Generator inertia	Jg	0.9×10^6 *N.m.s²*
Mechanical coupling damping coefficient	d	3.5×10^5 *N.m⁻¹.s*
Mechanical coupling stiffness coefficient	k	114×10^6 *N.m⁻¹*

Table 2 describes the four local models multimodel base used for the simulations.

Table 2: Multimodel base parameters.

Local model Mi	*Wind speed vi (m/s)*	*Pitch angle* $\beta_{i\text{-}nom}$ (°)
M1	*11.6*	*1.1*
M2	*14*	*8*
M3	*17*	*11.1*
M4	*25*	*15.4*

From this base, four optimal gains are calculated.

And thus, the stability feasibility problem consists in solving 8 LMI as shown after:

$$\begin{cases} P > 0 \quad (1\,LMI) \\ G_{ii}^{T}.P + P.G_{ii} < 0, i = 1..4 \quad (4\,LMI) \\ Q_{ij}^{T}.P + P.Q_{ij} < 0, i = 1..3,\ j = i+1 \quad (3\,LMI) \end{cases} \tag{33}$$

The simulation leads to the following result:

$$P = \begin{pmatrix} 8.848 & -8.335 & -0.008 & 0.298 \\ -8.335 & 8.101 & 0.007 & -0.258 \\ -0.53 & 0.523 & 0.026 & -0.001 \\ 0.298 & -0.258 & -0.001 & 0.084 \end{pmatrix}$$

Finding this positive definite matrix P is a sufficient condition proving the global stability of the control technique presented above.

For the simulations, we had chosen to place the closed loop poles for the local models at the left of -α=-0.5. This gives the following poles for each local model:

- *1st local model:*

$$P_1 = \begin{bmatrix} -1.032+12.24i \\ -1.032-12.24i \\ -1 \\ -1.069 \end{bmatrix}$$

- *2nd local model:*

$$P_2 = \begin{bmatrix} -1.032+12.24i \\ -1.032-12.24i \\ -1 \\ -1.06 \end{bmatrix}$$

- *3rd local model:*

$$P_3 = \begin{bmatrix} -1.033+12.24i \\ -1.033-12.24i \\ -1 \\ -1.049 \end{bmatrix}$$

- *4th local model:*

$$P_4 = \begin{bmatrix} -1.037+12.24i \\ -1.037-12.24i \\ -1 \\ -1.046 \end{bmatrix}$$

Thus, we can see that the pitch system pole (-1) is invariant for the four local models, and that all the other poles have their real parts less than –α.

To test the performance of this control strategy, a series of simulation for several wind steps has been performed to show the improvement of the studied controller against a classic multimodel LQ controller (Khezami *et al.*, 2009).

The Figure 6 presents a comparison simulation between the two control laws. For this simulation, only the turbine rotational speed response for a wind step of 0.5 m/s is presented for the four local models.

The local models poles for the classic LQ strategy have the following values:

- *1st local model:*

$$P_1 = \begin{bmatrix} -0.346+12.25i \\ -0.346-12.25i \\ -1.162+0.868i \\ -1.162-0.868i \end{bmatrix}$$

- *2nd local model:*

$$P_2 = \begin{bmatrix} -1.2+12.401i \\ -1.2-12.401i \\ -2.693+2.013i \\ -2.693-2.0128i \end{bmatrix}$$

- *3rd local model:*

$$P_3 = \begin{bmatrix} -1.62+12.587i \\ -1.62-12.587i \\ -3.417+2.287i \\ -3.417-2.287i \end{bmatrix}$$

- *4th local model:*

$$P_4 = \begin{bmatrix} -2.148+12.845i \\ -2.148-12.845i \\ -4.108+2.418i \\ -4.1081-2.418\text{i} \end{bmatrix}$$

In the comparison between both strategies, the system response for the proposed controller has indeed kept almost the same response time (about 4s), unlike the case with the classic LQ where the response times are varying from 12s for the 1st local model to 2s for the 4th local model.

Compared to the proposed strategy, the classic multimodel LQ control law shows responses with a more oscillating transient mode.

The improvements of the multimodel LQ controller consist in a more damped oscillatory mode and a faster dynamic than the classical control mode with an almost fixed response time for all the local models.

The simulation of the system with the proposed control strategy for a variable wind speed between 12m/s and 25m/s leads to the results presented in the curves of Figure 7.

The Figure 7 (a) illustrates a realistic aspect of the wind speed as described in a method elaborated by *C. Nichita* in (Nichita *et al.*, 2003). From this aspect, the controller allowed a good regulation of the generated electrical power (Figure 7 (d)) and the rotation speeds of both the rotor (Figure 7 (b)) and the generator (Figure 7 (c)) around their rated values with taking into account the fatigue damage since the mechanical torque (Figure 7 (e)) maintains an almost constant value which thereby leads to have alleviated mechanical loads.

The variations of the electromagnetic torque presented in Figure 7 (g) are smooth. However, the price paid for these performances is shown in Figure 7 (h) by a large activity of the pitch actuator.

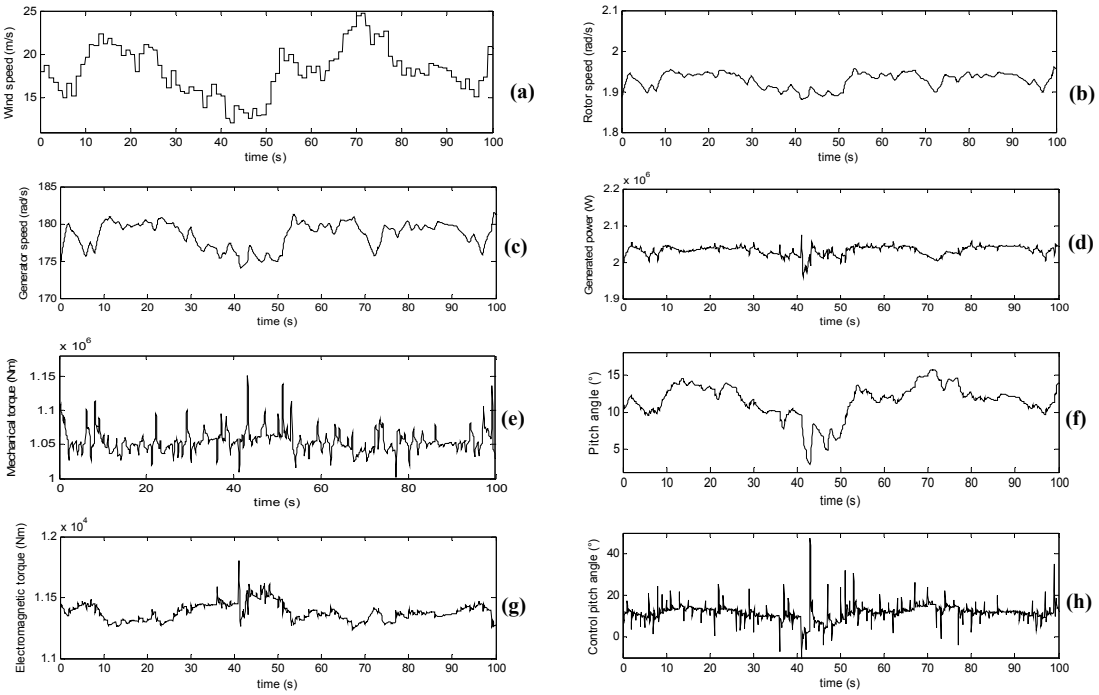


Figure 7: Variation of the system variables.

6 CONCLUSIONS

This paper dealt with a technique of designing a multimodel LQ regulator allowing to partially control the process global dynamic, and with a study of the global asymptotic stability of the controller by means of a set of LMI. The proposed strategy presented a compromise between different control objectives: optimizing the performances of the different system variables especially generating an electrical power of a good quality, minimizing the control efforts, alleviating the drive train dynamic loads and controlling the global dynamic of the studied process. The simulations results showed good performances of the controller with acceptable mechanical stress. But, satisfying such a trade-off between all these objectives is indeed difficult and the cost is however some high forces on the pitch actuator. These effects brought more challenges in the system analysis to improve the obtained results in order to control actively the system dynamic and to totally damp the oscillatory mode.

REFERENCES

Bianchi, F. D., Mantz, R. J. and Christiansen, C. F., 2004, Control of Variable-speed Wind Turbines by LPV Gain Scheduling. In *Wind Energy*, vol. 7, issue 1, pp:1-8.

Camblong, H., Rodriguez, M., Puiggali, J. R. and Abad, A., 2002, Comparison of different control strategies to study power quality in a variable-speed wind turbine. *In 1st World wind Energy Conference Proceeding, Berlin.*

Boukhezzara, B., Lupua, L., Siguerdidjanea, H. and Hand, M., 2007. Multivariable control strategy for variable speed, variable pitch wind turbines. In *Science Direct*, *Renewable Energy*, vol. 32, pp: 1273–1287.

Kardous, Z. , Benhadj Braiek, N. and Al Kamel, A., 2006. On the multimodel stabilization control of uncertain systems – Part 1, In *International Scientific IFNA-ANS Journal: Problems on Nonlinear Analysis in Engineering Systems,* No 2.

Kardous, Z., Benhadj Braiek, N. and Al Kamel, A., 2007. On the multimodel stabilization control of uncertain systems – Part 2. In *Int. J. of Problems of Nonlinear Analysis in Engineering Systems*, No.1(27), pp: 76-87, vol. 13.

Bianchi, F. D., De Battista, H. and Mantz, R. J., 2007. *Wind turbine control systems: principles, modeling and gain scheduling design*, Springer-Verlag. London, 1st edition.

Munteanu, I., Bratcu, A. I., Cultululis, N. A. and Ceanga, E., 2005. A two loop optimal control of flexible drive train variable speed wind power systems, *16th Triennial World Congress-IFAC*, Prague.

Khezami, N., Guillaud, X., and Benhadj Braiek, N., 2009. Multimodel LQ controller design for variable-speed and variable pitch wind turbines at high wind speeds. In *IEEE International Multi-conference on Systems, Signals and Devices*, Djerba.

Poulsen, N. K., Larsen, T. J., and Hansen, M. H., 2005. Comparison between a PI and LQ-regulation for a 2 MW wind turbine. *Risø National Laboratory-I-2320*.

Hammerum, K., Brath1, P. and Poulsen, N. K., 2007. A fatigue approach to wind turbine control. In *Journal of Physics*: Conference Series 75.

Cutululis, N. A., Bindner, H., Munteanu, I., Bratcu, A., Ceanga, E., and Soerensen, P., 2006. LQ Optimal Control of Wind Turbines in Hybrid Power Systems. In *European Wind Energy Conference and Exhibition*, Athens.

Chedli, M., 2002. *Stabilité et commande de systems décrits par des multimodèles*. PhD thesis, Institut National Polytechnique de Lorraine.

Liberzon, D. and Morse, A. S., 1999. Basic problems in stability and design of switched systems. In *IEEE Control Systems Magazine*, vol. 19, pp: 59-70.

Kardous, Z., Elkamel, A., Benhadj Braiek, N. and Borne, P., 2003. On the quadratic stabilization in discrete multimodel control. In *IEEE Conference on Control Applications*, vol.2, pp: 1398-1403.

Nichita, C., Luca, D. and Dakyo, B., 2003. Méthodes de simulation de la vitesse du vent. In *Decentralized energy seminar,* Toulouse.

[*] *This work was supported by the CMCU project number:* ***08G1120***

MINIMIZING THE MAKESPAN IN TWO-MACHINE JOB SHOP SCHEDULING PROBLEMS WITH NO MACHINE IDLE-TIME

Fatma Hermès
Faculté des Sciences Mathématiques, Physiques et Naturelles de Tunis, Département Informatique, Laboratoire LI3
Campus Universitaire, 1060 Tunis, Tunisia
fatma.hermes@fst.rnu.tn

Jacques Carlier, Aziz Moukrim
Université de Technologie de Compiègne, Centre de Recherches de Royallieu
Laboratoire Heudiasyc, UMR CNRS 6599, BP 20529, 60205 Compiègne cedex, France
{jacques.carlier, aziz.moukrim}@hds.utc.fr

Khaled Ghédira
Institut Supérieur de Gestion de Tunis, Laboratoire LI3
Université de Tunis, 41 Rue de la Liberté – Bouchoucha, 2000 Bardo, Tunisia
khaled.ghedira@isg.rnu.tn

Keywords: Scheduling, Job Shop, Two Machines, No-idle Constraint, Makespan, Optimal Solution.

Abstract: This paper deals with two-machine job shop scheduling problems working under the no-idle constraint, that is, machines must work continuously without idle intervals. The makespan (C_{max}) has to be minimized. First, we study the problem where each job consists of at most two operations and we show that it can be solved polynomially using Jackson's rule (Jackson, 1956). Second, we study the problem where the number of operations per job can be greater than two and all operations are of unit time and we extend the results of (Hefetz and Adiri, 1982). Finally, we discuss the possibility of getting feasible solutions and then optimal solutions in the general case where the number of operations per job can be greater than two and all operations do not have the same processing time.

1 INTRODUCTION

Frequently, the cost of making machines wait is so high that a no-idle constraint is imposed on machines and no intermediate idle time between operations processed by the same machine is allowed. For example, if the machine is an oven that must cook some pieces at a given high temperature then maintaining the required temperature of the oven while it is empty may be too costly. However, studies of problems on this topic have not attracted a great deal of attention. In the literature, we find some works, most of which are recent, on the permutation flow shop ((Adiri and Pohoryles, 1982), (Baptiste and Lee, 1997), (Kalczynski and Kamburowski, 2007), (Saadani, Guinet and Moalla, 2001), (Saadani, Guinet and Moalla, 2003)). There are also some recent works discussing one machine scheduling problems ((Chrétienne, 2008), (Valente and Alves, 2005), (Valente, 2006)).

The aim of this paper is to study two-machine job shop problems where a set I of n jobs, I = {1, ..., n}, has to be scheduled without intermediate delay on two machines in order to minimize the maximum of the completion times of the jobs i.e. the makespan (C_{max}). Each job i, $i \in I$, is composed of n_i operations $O_{i,j}$, $j = 1 \ldots n_i$, and each operation $O_{i,j}$ has to be processed on a fixed machine for $p_{i,j}$ time units.

The job shop problem plays an important role in the scheduling theory because of its practical applications. Most of job shop problems are NP-hard and there are only few special cases which can be solved polynomially. The two-machine job shop problem with at most two operations per job is denoted $J2|n_i \leq 2|C_{max}$. It was solved polynomially by Jackson (Jackson, 1956) who proposed an algorithm which calculates an optimal schedule in O(n*log(n)) steps using Johnson's rule (Johnson, 1954). The two-machine unit-time job shop problem is denoted

$J2|p_{i,j}=1|C_{max}$. It was proved to be polynomial by (Hefetz and Adiri, 1982) where authors proposed the longest remaining processing time first algorithm which schedules operations in a decreasing order of the remaining processing time of jobs. Lenstra, Rinnooy Kan and Brucker showed in (Lenstra, Rinnooy Kan and Brucker, 1977) that problem $J2||C_{max}$ is strongly NP-hard. Later, Brucker showed in (Brucker, 1994) that the two-machine job shop problem with a fixed number k of jobs, denoted $J2|n=k|C_{max}$, can be solved polynomially by reducing it to a shortest path problem and then he deduced that it is possible to calculate an optimal schedule for $J2||C_{max}$ for any fixed number of jobs in polynomial time.

The paper is organized as follows: In section 2, we define the studied problem and we recall some backgrounds and basic results relying on the two-machine job shop problem. In section 3, we study the two-machine job shop problem where the number of operations per job is at most equal to two and machines must work under the no-idle constraint. This problem is denoted $J2|n_i\leq 2,\text{no-idle}|C_{max}$. We show that it can be solved in polynomial time using Jackson's rule (Jackson, 1956). In section 4, we study the two-machine unit-time job shop problem with no machine idle time, denoted $J2|p_{i,j}=1,\text{no-idle}|C_{max}$, and we extend the results of (Hefetz and Adiri,1982). Finally, in section 5, we deduce some special cases which are polynomially solvable.

2 GENERAL POINTS

In this section, we first define the problem subject of this study and we present some definitions. Next, we present Johnson's and Jackson's algorithms where Johnson's algorithm (Johnson, 1954) solves the two-machine flow shop problem with C_{max} criterion denoted $F2||C_{max}$ and Jackson's algorithm (Jackson, 1956) solves problem $J2|n_i\leq 2|C_{max}$ using Johnson's rule. Finally we introduce the longest remaining processing time first algorithm (Hefetz and Adiri, 1982) which solves problem $J2|p_{ij}=1|C_{max}$.

2.1 Problem Formulation and Basic Definitions

The two-machine job shop problem is a problem where a set I of n jobs, $I = \{1, ..., n\}$ have to be processed in a shop with two machines M_1 and M_2. Each job i, $i = 1, \ldots, n$, consists of a sequence of n_i operations $O_{i,1}, O_{i,2}, ..., O_{i,ni}$ which must be processed in this order. The precedence constraints are so that $O_{i,j}$ precedes $O_{i,j+1}$, $j = 1, \ldots, n_i - 1$. Each operation $O_{i,j}$ must be processed for $p_{i,j}$ time units on machine $\mu_{i,j} \in \{M_1, M_2\}$.

The following assumptions are made:

- A machine can process only one operation at a time.
- An operation cannot be interrupted.
- The time zero is the earliest time an operation can be started.
- All setup times are included into the job processing times.
- If operation $O_{i,j}$ must be processed on machine M_1, then operation $O_{i,j+1}$ must be processed on machine M_2 ($\mu_{i,j} \neq \mu_{i,j+1}$ for $i = 1,\ldots, n_i - 1$). Thus, job i may be characterized by the number of operations and the machine on which the first operation must be processed.
- Only no-idle schedules are considered.

Let $t_{i,j}$ be the starting time of operation $O_{i,j}$ and let $C_{i,j}$ be its completion time. Let C_i be the completion time of job i so that

$$C_i = \max_{j=1\ldots n_i} C_{i,j} \quad (1)$$

Let us present the following definitions:

- An *initial operation* is one without predecessors: the operation $O_{i,1}$ is the initial operation for job i.
- A *terminal operation* is one without successors: the operation O_{i,n_i} is the terminal operation for job i.
- A *ready operation* is an operation that has not yet been scheduled while all its predecessors have been.
- A *no-idle schedule* satisfies the no-idle constraint on each machine. In other words, if operation $O_{i,j}$ is executed immediately before operation $O_{i',j'}$ on the same machine then we have:

$$C_{i,j} = t_{i,j} + p_{i,j} = t_{i',j'} \quad (2)$$

Given a feasible schedule π we have:

$$C_{max}(\pi) = \max_{\substack{i=1\ldots n \\ j=1\ldots n_i}} C_{i,j} = \max_{i=1\ldots n} C_i \quad (3)$$

The objective is to find a no-idle schedule so as to minimize the C_{max}. The problem so formulated is denoted $J2|\text{no-idle}|C_{max}$. When the number of operations per job is at most equal to two, the problem is denoted $J2|n_i\leq 2,\text{no-idle}|C_{max}$ and when all operations have the same processing time which is considered as the unit-time, the problem is denoted $J2|p_{i,j}=1,\text{no-idle}|C_{max}$.

2.2 Jackson's and Johnson's Algorithms

Jackson's algorithm constructs an optimal schedule for $J2|n_i \leq 2|C_{max}$ problem reducing it to $F2||C_{max}$ problem. Below, we first describe Johnson's algorithm. Then, we present Jackson's algorithm. Finally, we conclude with some results concerning the presence of idle times on machines.

2.2.1 Johnson's Algorithm

The two-machine flow shop problem is a problem where a set I' of n' jobs, $I' = \{1,2,\ldots,n'\}$ must visit machines in the same order. Each job i consists of two operations $O_{i,1}$ and $O_{i,2}$ which must be processed respectively, first on machine M_1 then on machine M_2. Johnson's algorithm (Johnson, 1954) constructs an optimal schedule in polynomial time ($O(n*\log(n))$) for $F2||C_{max}$ problem. It applies the following steps:

i. Divide the set of jobs I', $I' = \{1,2,\ldots,n'\}$, into two subsets:
   a. Let I'_1 denote the subset of jobs i, i = 1, …, n', which satisfy the condition $p_{i,1} \leq p_{i,2}$
   b. Let I'_2 denote the subset of jobs i, i = 1, …, n', which satisfy the condition $p_{i,1} > p_{i,2}$
ii. Schedule on each machine first the jobs of I'_1 in an increasing order of $p_{i,1}$ and then the jobs of I'_2 in a decreasing order of $p_{i,2}$.

2.2.2 Jackson's Algorithm

Jackson's algorithm (Jackson, 1956) calculates an optimal solution for problem $J2|n_i \leq 2|C_{max}$, in polynomial time ($O(n*\log(n))$) by first reducing it to $F2||C_{max}$ problem and then using Johnson's rule. It applies the following steps:

i. Divide the set of jobs I = {1, 2, …, n} into four subsets:
   a. Let I_1 denote the subset of jobs consisted of only one operation which must be processed on machine M_1.
   b. Let I_2 denote the subset of jobs consisted of only one operation which must be processed on machine M_2.
   c. Let $I_{1,2}$ denote the subset of jobs which are processed first on machine M_1 then on machine M_2.
   d. Let $I_{2,1}$ denote the subset of jobs which are processed first on machine M_2 then on machine M_1.
ii. Calculate an optimal sequence $R_{1,2}$ for the flow shop problem relative to the job set $I_{1,2}$.
iii. Calculate an optimal sequence $R_{2,1}$ for the flow shop problem relative to the job set $I_{2,1}$.
iv. On machine M_1 schedule first $I_{1,2}$ according to $R_{1,2}$, then all jobs in I_1 and finally $I_{2,1}$ according to $R_{2,1}$.
v. On machine M_2 schedule first $I_{2,1}$ according to $R_{2,1}$, then all jobs in I_2 and finally $I_{1,2}$ according to $R_{1,2}$.

2.2.3 Further Results

Adiri and Pohoryles (Adiri and Pohoryles, 1982) observe that problems $F2|prmu,no\text{-}idle|C_{max}$ and $F2|prmu|C_{max}$ are equivalent in the sense that every $F2|prmu|C_{max}$ schedule can be transformed into an $F2|prmu,no\text{-}idle|C_{max}$ schedule with maintaining the same C_{max}. Thus, both problems can be solved by Johnson's algorithm (Johnson, 1954). Johnson's schedule is an active schedule in which machine M_1 is naturally no-idle since operations are processed consecutively on it without idle interval. Moreover, the jobs preceding each idle interval on the second machine M_2 can be delayed without increasing the C_{max}. It is enough to fix the starting time of the last operation scheduled on M_2 and to schedule the other operations so that all operations are scheduled consecutively without any intermediate delay.

Brucker (Brucker, 1995) observes that in Jackson's schedule at least one machine processes jobs without idle intervals. More specifically, having the following assumption:

$$\sum_{i \in I_{2,1}} p_{i,2} \leq \sum_{i \in I_{1,2}} p_{i,1} + \sum_{i \in I_1} p_{i,1} \qquad (4)$$

then there is no idle time on machine M_1. Otherwise, there is no idle time on machine M_2.

2.3 The Longest Remaining Processing Time First Algorithm

The longest remaining processing time first algorithm has been proposed by (Hefetz and Adiri, 1982) to solve $J2|p_{i,j}=1|C_{max}$ problem. It constructs an optimal schedule for this problem with applying the following steps:

i. Give a label $\alpha_{i,j}$ to each operation $O_{i,j}$ so that:

$$\alpha_{i,j} = n_i - j + 1 \quad i = 1,\ldots,n \text{ and } j = 1,\ldots,n_i \qquad (5)$$

ii. Schedule the highest label operation for the earliest possible time on the required machine, with ties broken arbitrarily.
iii. Remove from the problem the scheduled operation. Stop if all operations are scheduled, otherwise return to (ii).

The authors noted that the operation with the highest label in step (ii) must be a ready operation,

since, if it is not then there is an unscheduled predecessor with a higher label, which is a contradiction.

Let T_j, j = 1, 2, be the total processing time required on machine M_j and p_i the processing time of job i. Thus, $p_i = n_i$ in view of the fact that all operations have unit duration.

Theorem 1 (Hefetz and Adiri, 1982). The longest remaining processing time first algorithm constructs an optimal schedule for $J2|p_{i,j}=1|C_{max}$ problem. If all initial operations require the same machine and we have:

$$T_1 = T_2 \geq \max_i n_i \quad (6)$$

Then, the optimal schedule length is:

$$C^*_{max} = T_1 + 1 = T_2 + 1 \quad (7)$$

Otherwise, the optimal schedule length is:

$$C^*_{max} = \max(T_1, T_2, \max_i n_i) \quad (8)$$

3 RESOLUTION OF THE PROBLEM $J2|N_I \leq 2, NO\text{-}IDLE|C_{MAX}$

The problem we consider in this section is to find an optimal schedule for $J2|n_i \leq 2, no\text{-}idle|C_{max}$ problem. We propose first to study the feasibility of Jackson's schedule then, we prove its optimality. We also deduce some interesting results which concern the necessity of applying Johnson's rule in Jackson's algorithm.

Proposition 1. The C_{max}-value of Jackson's schedule is a lower bound for the optimal C_{max} of $J2|n_i \leq 2, no\text{-}idle|C_{max}$ problem.

Proof. Jackson's schedule is an active schedule in which all the operations are scheduled as soon as possible. On the other hand, to satisfy the no-idle constraint, some operations must be delayed which must increase the C_{max}-value. So, the optimal C_{max} of $J2|n_i \leq 2, no\text{-}idle|C_{max}$ problem must be greater than or equal to that of $J2|n_i \leq 2|C_{max}$ problem.

So, evidently Jackson's schedule is optimal for $J2|n_i \leq 2, no\text{-}idle|C_{max}$ problem if it is no-idle or if it can be transformed into a no-idle schedule without increasing the C_{max}-value.

Lemma 1. If (4) then there is no-idle time on machine M_1 and it is unnecessary for Jackson's algorithm to apply Johnson's rule for the two-machine flow shop sub-problem relative to $I_{2,1}$.

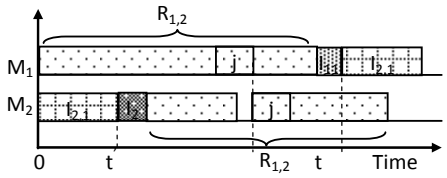


Figure 1: A schedule where there is no-idle time on machine M_1.

Proof. If (4) holds, then at time t where,

$$t = \sum_{i \in I_{1,2}} p_{i,1} + \sum_{i \in I_1} p_{i,1} \quad (9)$$

all the operations of $I_{2,1}$ which must be processed on machine M_1 are ready. Therefore, these operations can be processed without idle time immediately after the end of those of I_1 as presented in figure 1 above.

As a result, the order of processing the operations of $I_{2,1}$ on machine M_1 and on machine M_2 does not affect the C_{max}-value since all the operations of $I_{2,1}$ which must be processed on machine M_2 are naturally ready at time 0. They are completed at time t', where

$$t' = \sum_{i \in I_{2,1}} p_{i,2} \leq t \quad (10)$$

Lemma 2. If

$$\sum_{i \in I_{1,2}} p_{i,1} \leq \sum_{i \in I_{2,1}} p_{i,2} + \sum_{i \in I_2} p_{i,2} \quad (11)$$

then there is no-idle time on machine M_2 and it is unnecessary for Jackson's algorithm to apply Johnson's rule for the two-machine flow shop sub-problem relative to $I_{1,2}$.

Proof. The proof is similar to that of lemma 1.

Let us note that if

$$\sum_{i \in I_{2,1}} p_{i,2} \leq \sum_{i \in I_{1,2}} p_{i,1} \quad (12)$$

then (4) holds. Also, if

$$\sum_{i \in I_{1,2}} p_{i,1} \leq \sum_{i \in I_{2,1}} p_{i,2} \quad (13)$$

then (11) holds.

By summation of (12) and (13), we have:

$$\sum_{i \in I_{1,2}} p_{i,1} = \sum_{i \in I_{2,1}} p_{i,2} \quad (14)$$

Lemma 3. If any of the following assumptions hold, namely, (4) and (13), (11) and (12), or (14) then Jackson's schedule is no-idle. It is consequently

optimal for problem $J2|n_i\leq 2,\text{no-idle}|C_{max}$ and it is unnecessary for Jackson's algorithm to apply Johnson's rule neither for the two-machine flow shop sub-problem relative to $I_{1,2}$ nor for the two-machine flow shop sub-problem relative to $I_{2,1}$.

Proof. If (4) holds then there is no idle time on machine M_1 and it is pointless to apply Johnson's rule in Jackson's algorithm to the sub-problem relative to $I_{2,1}$ (lemma 1).

If (13) holds then there is no-idle time on machine M_2 and it is pointless to apply Johnson's rule in Jackson's algorithm to the sub-problem relative to $I_{2,1}$ (lemma 2).

By summation of (4) and (13), Jackson's schedule is no-idle. It is consequently optimal for $J2|n_i\leq 2,\text{no-idle}|C_{max}$ problem and it is unnecessary for Jackson's algorithm to apply Johnson's rule neither for the two-machine flow shop sub-problem relative to $I_{1,2}$ nor for the two-machine flow shop sub-problem relative to $I_{2,1}$.

In the same way, we deduce the same results if assumptions (11) and (12) hold or assumption (14) holds.

Proposition 2. The problem $J2|n_i\leq 2,\text{no-idle}|C_{max}$ is polynomial. It can be solved with Jackson's algorithm.

Proof. At least one machine is no-idle (Brucker, 1995). If there is an idle interval on machine M_1 then it is between the operations of $I_{2,1}$ and in this case the assumption (4) does not hold. If there is an idle interval on machine M_2 then it is between the operations of $I_{1,2}$ and in this case the assumption (11) does not hold either.

Supposing that machine M_2 contains an idle time, then this idle time is between the operations of $I_{1,2}$ relative to the two-machine flow shop sub-problem where each job visits first, machine M_1 then machine M_2. This idle time can be reduced by fixing the starting time of the last operation scheduled on machine M_2 and scheduling the other operations of $I_{1,2}$ consecutively without any intermediate delay (Adiri and Pohoryles, 1982).

This action creates an other idle time between the first operation scheduled of $I_{1,2}$ on machine M_2 and the last operation scheduled of I_2 on the same machine.

The last idle time can also be reduced by delaying all the operations of I_2 and then the idle time becomes between the last operation scheduled of $I_{2,1}$ and the first operation scheduled of I_2. Since, there is no idle time on machine M_1, then (4) holds.

Let t be the ending date of the last operation of $I_{1,2}$ scheduled on machine M_1 and let t' be the starting time of the first operation of $I_{1,2}$ scheduled on machine M_2.

Naturally, we have:

$$t' \leq t \qquad (15)$$

Because if (15) does not hold, then (4) does not hold. Consequently, it is possible to shift the operations of $I_{2,1}$ on machine M_2 to the right so that there is no-idle time on machine M_2.

Thus, the schedule constructed by Jackson's algorithm can be easily transformed into a no-idle schedule without increasing the C_{max}-value. It is enough to fix the last operation on the machine which contains idle intervals and shift to the right all the other operations in order to have no idle intervals.

As a result, Jackson's algorithm also constructs an optimal schedule for he problem $J2|n_i\leq 2,\text{no-idle}|C_{max}$ problem.

Proposition 3. The set of optimal solutions of $J2|n_i\leq 2,\text{no-idle}|C_{max}$ problem is included in the set of optimal solutions of $J2|n_i\leq 2|C_{max}$ problem.

Proof. Jackson's schedule can easily be transformed to a no-idle schedule without increasing the C_{max}-value. So, both problems have the same C_{max}^*. Besides, an optimal solution for $J2|n_i\leq 2,\text{no-idle}|C_{max}$ problem is also optimal for $J2|n_i\leq 2|C_{max}$ problem.

Thus, the set of optimal solutions of $J2|n_i\leq 2,\text{no-idle}|C_{max}$ problem is included in the set of optimal solutions of $J2|n_i\leq 2|C_{max}$ problem.

4 RESOLUTION OF THE PROBLEM $J2|p_{ij}=1,\text{no-idle}|C_{max}$

In this section, we study $J2|p_{ij}=1,\text{no-idle}|C_{max}$ problem.

Proposition 4. The optimal C_{max} of $J2|p_{ij}=1|C_{max}$ problem is a lower bound for the optimal C_{max} of $J2|p_{ij}=1,\text{no-idle}|C_{max}$ problem; if an optimal schedule exists.

Proof. The proof is similar to that of proposition 1.

Below, we denote HA the longest remaining processing time first algorithm. Let S be the schedule constructed by HA algorithm.

Lemma 4. If there is no-idle time in S then the makespan of S is minimal and the number of idle times is minimal.

Proof. HA calculates a schedule with minimal makespan for $J2|p_{ij} = 1|C_{max}$ problem (theorem 1). Moreover, if there is no-idle time in S then the number of idle times is also minimal.

Let us introduce a second criterion to minimize which is the number of idle times denoted $\bar{A}$. The objective is then to minimize first the C_{max} then $\bar{A}$. In this case, the related problem is denoted $J2|p_{ij}=1|Lex(C_{max},\bar{A})$.

Evidently, if an optimal schedule for $J2|p_{ij}=1|Lex(C_{max},\bar{A})$ problem is no-idle then it is also optimal for $J2|p_{ij}=1,no\text{-}idle|C_{max}$ problem. Also, if there is no-idle time in S, then S is optimal for both problems $J2|p_{ij}=1,no\text{-}idle|C_{max}$ and $J2|p_{ij}=1|Lex(C_{max},\bar{A})$. Furthermore, if it is possible to built a no-idle schedule S' from S without increasing the C_{max} then S' is optimal for both problems $J2|p_{ij}=1,no\text{-}idle|C_{max}$ and $J2|p_{ij}=1|Lex(C_{max},\bar{A})$.

Let us assume that job h is the job so that

$$n_h = \max_i n_i \quad (16)$$

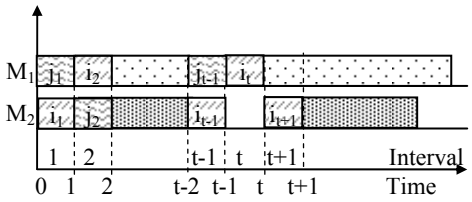


Figure 2: The schedule resulting from the assumption of an idle interval on machine M_2 in interval t and t is even.

Lemma 5. If in S there is an idle time, on some machine, then

i. the job h is unique, and it is processed continuously from time 0 to n_h, alternating on the two machines;
ii. the only job processed on the machine containing this idle time is the job h;
iii. from time t – 1, all the operations processed on the other machine, except of those of the job h, must have a label 1.

Proof. The time axis is supposed to be split into intervals of unit times. We suppose that the first idle interval is the interval t. Let us suppose also that t is even and that this idle time is on machine M_2. Let us denote i_1 (resp: j_1) the job processed in interval 1 on machine M_2 (resp: M_1) and so on. So, i_{t-1} is the job processed on machine M_2 in interval t – 1, and i_t is the job processed on machine M_1 in interval t (figure 2). The label of job j_{t-1} is 1. Otherwise there doesn't exist any idle time in interval t. Consequently, all jobs processed after t – 1, at the exception of job i_t which is the only job which can have a label strictly larger than 1. Let us set $i = i_t$. We prove that $i_{t-1} = i$. It is the only job which can have a label strictly larger than 2. For the same reason $i_{t-2} = i$ (the only job which can have a label strictly larger than 3) and so on until $i_1 = i$. Consequently, i = h. The same reasoning can be applied when t is odd.

Thus, the job i must be the job h and then the job h is continuously scheduled from beginning to end since at each time it have the greatest label. It is consequently the job h verifying the assumption (16) and it is unique.

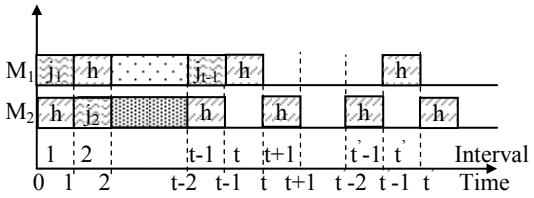


Figure 3: The first operation of job h is processed on machine M_2 and machine M_1 work for the first interval.

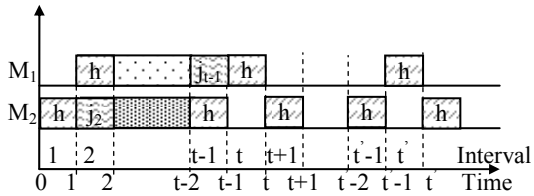


Figure 4: The first operation of job h is processed on machine M_2 and machine M_1 is free for the first interval.

Algorithm IT

i. Built S with applying HA algorithm
ii. Built S' by scheduling the first operation of job h at time 0 and letting the other machine idle at time 0 and scheduling the remaining jobs by applying HA algorithm from time 1.

Theorem 2. One of the two schedules built by algorithm IT is optimal for both objective functions.

Proof. First, let us note that if there is an idle time in S then this idle time usually precedes an operation of job h and it is also preceded with an operation of the same job h. Indeed, if there is an idle time then the job h is unique and therefore it is continuously scheduled from beginning to the end (lemma 5). Thus, in each interval the job h is processed on one machine and on the other machine either another job is processed or there is an idle time. On the other hand, if the first operation of job h is processed on

machine M_2 then if there is an idle time on machine M_2 which occurs in interval t then t is even and if there an idle time on machine M_1 which occurs in interval t' then t' is odd. Otherwise, t is odd and t' is even.

Below, we show in four cases that S or S' is optimal for both objectives.

In the first case, we assume that the first operation of job h is processed on machine M_2 and the machine M_1 works for the first interval of time (figure 3). In this case, the number of idle times on machine M_2 is minimal because the operations scheduled on machine M_2, except of those of the job h, are used optimally to fill idle intervals created by the job h. It is the same for machine M_1 from time 1. However, if it is possible to make machine begin working later with one unit of time without increasing the ending date on this machine and the C_{max} then we can suppress an idle time. This is done eventually in S'.

In the second case, we assume that the first operation of job h is processed on machine M_1 and the machine M_2 works for the first interval of time. This case is similar to the first one.

In the third case, we assume that the first operation of job h is processed on machine M_2 and the machine M_1 is free for the first interval of time (figure 4). In this case, the number of idle times is minimal in both machines because in each machine the first operation scheduled is an operation of job h.

In the fourth case, we assume that the first operation of job h is processed on machine M_1 and the machine M_2 is free for the first interval of time. This case is similar to the third one.

Thus, S or S' is optimal for both objectives.

We can resume that S is no-idle if and only if one of the following cases holds:

Case 1:

$$\max_i n_i \leq \min(T_1, T_2) \quad (18)$$

Case 2:

$$\max_i n_i = \min(T_1, T_2) + 1 \quad (19)$$

and the last operation of the job h is scheduled on the machine which determines the schedule length.

Case 3:

$$\max_i n_i = \min(T_1, T_2) + 2 \quad (20)$$

and all jobs begin in the same machine and the last operation of the job h is scheduled on the machine which determines the schedule length.

Finally, we note that S' is no-idle if S is no-idle or S contains only one idle time.

5 RESOLUTION OF THE PROBLEM J2|n=k, no-idle|C_{max}

In this section, we discuss J2|n=k,no-idle|C_{max} problem.

Proposition 5. The optimal C_{max} of J2|n=k|C_{max} problem is a lower bound for the optimal C_{max} of J2|n=k,no-idle|C_{max} problem; if an optimal schedule exists.

Proof. The proof is similar to that of proposition 1.

So, the optimal schedule for J2|n=k|C_{max} problem is optimal for J2|n=k,no-idle|C_{max} problem if it is no-idle or if it can be transformed into no-idle schedule without increasing the C_{max}-value.

Obviously, if n = 1 and $n_1 > 2$ then it is not possible to construct a no-idle schedule. There are exactly $n_1 - 2$ idle intervals.

Let us consider the problem where the number of jobs is equal to two. This problem is denoted J2|n=2,no-idle|C_{max}.

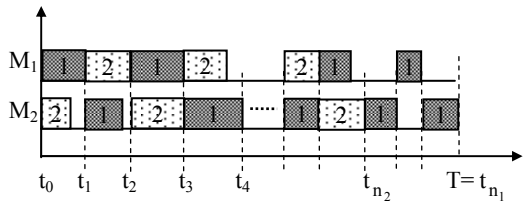


Figure 5: Schedule format with two jobs where $n_1 > n_2$.

Let us give the following cases:

Case 1: $n_1 = n_2$, $\mu_{1,1} \neq \mu_{2,1}$ and $p_{1,j} = p_{2,j}$ for $j = 2 \ldots n_1 - 1$

Case 2: $n_1 = n_2$, $\mu_{1,1} = \mu_{2,1}$ and $p_{1,j+1} = p_{2,j}$ for $j = 1 \ldots n_1 - 1$

Case 3: $n_1 = n_2$, $\mu_{1,1} = \mu_{2,1}$ and $p_{1,j} = p_{2,j+1}$ for $j = 1 \ldots n_1 - 1$

Case 4: $n_1 = n_2 + 1$, $\mu_{1,1} \neq \mu_{2,1}$ and $p_{1,j} = p_{2,j}$ for $j = 2 \ldots n_2$

Case 5: $n_1 = n_2 + 1$ and $\mu_{1,1} = \mu_{2,1}$ and $p_{1,j+1} = p_{2,j}$ for $j = 1 \ldots n_2 - 1$

Case 6: $n_2 = n_1 + 1$, $\mu_{1,1} \neq \mu_{2,1}$ and $p_{1,j} = p_{2,j}$ for $j = 2 \ldots n_1$

Case 7: $n_2 = n_1 + 1$, $\mu_{1,1} = \mu_{2,1}$ and $p_{1,j} = p_{2,j+1}$ for $j = 1 \ldots n_1 - 1$

Case 8: $n_1 = n_2 + 2$, $\mu_{1,1} = \mu_{2,1}$, and $p_{1,j+1} = p_{2,j}$ for $j = 1 \ldots n_2$

Case 9: $n_2 = n_1 + 2$, $\mu_{1,1} = \mu_{2,1}$, and $p_{1,j} = p_{2,j+1}$ for $j = 1 \ldots n_1$

Proposition 6. If any of the previous cases holds then the no-idle schedule is unique; it is consequently the optimal solution for J2|n=2,no-idle|C_{max} problem. Otherwise, it is impossible to get a no-idle schedule.

Proof. A feasible schedule for J2|n=2|C_{max} problem takes the format presented in figure 5 above. Clearly, having this format, it is not possible to transform any schedule to a no-idle schedule. There are at most $\max(n_1, n_2) - 2$ idle intervals.

However, we deduce that if any of the previous cases holds then the no-idle schedule is unique. It is consequently the optimal solution for J2|n=2,no-idle|C_{max} problem. Otherwise, it is impossible to get a no-idle schedule.

6 CONCLUSIONS

In this paper, we have studied the impact of adding the no-idle constraint to the problem of minimizing the makespan in a two-machine job shop. We have studied separately the case where the number of operations per job isn't greater than two and the case where all operations are of unit time. In the first case we have showed that there exists usually an optimal schedule which we can calculate using Jackson's rule and then fixing the last operation scheduled on the machine which contains an idle time and then scheduling the other operations consecutively without idle times. However, in the second case, we showed that it is not usually possible to build a feasible no-idle schedule. Then, we have proposed the IT algorithm which minimizes first the C_{max} then the number of idle times ($\bar{A}$). We have shown that it is impossible to build a schedule which contains a number of idle times smaller than that of the schedule obtained by applying IT algorithm. Consequently, if this schedule is no-idle then it is also optimal for the corresponding problem with adding the no-idle constraint. Moreover, in the general case, where the number of operations per job can be greater than two and all operations do not have the same processing time, we have shown that where the number of jobs is equal to two there are only few cases numbered from 1 to 9 which are efficiently solvable and where the set of feasible no-idle schedules contains a unique schedule. In conclusion, we deduce that it is not usually possible to construct a feasible no-idle schedule for the two-machine job shop problem and that in the majority of cases, this set is empty.

REFERENCES

Adiri, I., Pohoryles, D., 1982. Flow-shop/no-idle or no-wait scheduling to minimise the sum of completion times. *Naval Research Logistics Quarterly*, Vol. 29, pp. 495-504.

Baptiste, P., Lee, K.H., 1997. A branch and bound algorithm for the F|no-idle|C_{max}. *Proceedings of the International Conference on Industrial Engineering and Production Management (IEPM'1997)*, Lyon, vol. 1, pp. 429-438.

Brucker, P., 1994. A polynomial algorithm for the two machine job-shop scheduling problem with fixed number of jobs. *Operations Research Spektrum*, Vol. 16, pp. 5-7.

Brucker, P., 1995. Scheduling Algorithms, *Springer*, ISBN: 3-540-60087-6.

Chrétienne, P., 2008. On single-machine scheduling without intermediate delays. *Discrete Applied Mathematics, 156,* p. 2543-2550.

Garey, M.R.D., Johnson, D.S., Sethi, R., 1976. The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, Vol. 1, pp. 117-129.

Hefetz, N. , Adiri, I., 1982. An Efficient Optimal Algorithm for the Two-Machines Unit-Time Jobshop Schedule-Length Problem. *Mathematics of Operations Research*, Vol. 7, No. 3., pp. 354-360.

Jackson, J.R., 1956. An extension of Johnson's results on job lot scheduling. *Naval Research Logistic Quarterly*, Vol. 3, pp. 201-203.

Johnson, S.M., 1954. Optimal two- and three-stage production schedules with setup times included, *Naval Research Logistics Quarterly*, Vol. 1, pp. 61-68.

Kalczynski, P. J., Kamburowski, J., 2007. On no-wait and no-idle flow shops with makespan criterion. *European Journal of Operational Research*, Vol. 178, pp. 677–685.

Lenstra, J. K., Rinnooy Kan, A.H.G., Brucker, P., 1977. Complexity machine scheduling problems. *Annals of Discrete Mathematics*, Vol. 1, pp. 343-362.

Saadani, H., Guinet, A., Moalla, M., 2001. A travelling salesman approach to solve the *F*|no-idle|C_{max} problem. *Proceedings of the International Conference on Industrial Engineering and Production Management (IEPM' 2001)*, Quebec, vol. 2, 880-888.

Saadani, H., Guinet, A., Moalla, M., 2003. Three stage no-idle flow-shops. *Computers and Industrial Engineering*, 44, 425-434.

Valente, J. M. S., Alves, R. A. F. S., 2005. Improved Heuristics for the Early/Tardy scheduling problem with no idle time. *Computers & Operations Research*, 32: p. 557-569.

Valente, J. M. S., 2006. Heuristics for the single machine scheduling problem with early and quadratic tardy penalities. Working paper 234, Faculdade de Economia do Porto, Portugal, December.

A LOW COST AND FLEXIBLE APPROACH TO CAN CONFORMANCE TESTING

Imran Sheikh and Michael Short
Embedded Systems Laboratory, Department of Engineering, University of Leicester, University Road, Leicester, U.K.
{si52, mjs61}@le.ac.uk

Keywords: Controller Area Network, Conformance testing, Network protocol verification.

Abstract: Since its introduction in the early 1980's, CAN has become the de-facto communications protocol employed in vehicle and industrial control applications. Before any new product can claim to support CAN-connectivity, compliance with the protocol at the physical and data link layers must be tested and verified. To help standardize the requirements for such testing, ISO has set a draft standard specifically for CAN conformance testing. Traditionally, CAN controllers and transceivers have been implemented at the silicon level, either in the form of dedicated IC's or as on-chip peripherals of embedded devices. The practical implementation of CAN conformance testers has been realised using dedicated hardware and specially written analysis software; this is a practical approach when testing and verifying conformance prior to high-volume IC manufacture. However, recent years have seen an increased interest in the employment of CAN-connected devices implemented by programmable logic devices such as FPGA's. Such 'soft core' implementations are often in small-volume (or even one-off) batches. In such circumstances, for cost and availability reasons, it may not be practical for developers to use traditional CAN-conformance testing equipment. To help alleviate this problem, this paper proposes a low-cost and easily implemented method which will allow developers to fully test a CAN soft core implementation. The method is based around simple off-the-shelf development boards and the simple analysis tool Chipscope, and allows developers to verify a CAN core against the relevant ISO standards. Finally, the paper describes the use of the test bed in the verification of an open-source CAN soft core implementation.

1 INTRODUCTION

Conformance testing is an integral part of the development stage of any network protocol implementation. When components (or devices) pass such conformance tests this ensures, to an acceptable degree of confidence, that the implementation of the given set of protocol specifications has been correctly interpreted by the designers; and it also has been instantiated in a form that is free from errors.

Since its introduction in the early 1980's, the Controller Area Network (CAN) protocol has become the de-facto communications protocol employed in vehicle and industrial control applications (Bosch, 1991). In light of the popularity of CAN, the ISO has developed a standard exclusively aimed at CAN conformance testing. Before any new equipment design can claim to be CAN conformant, evidence is required that shows that the testing procedures outlined in ISO 16845 (ISO, 2000) have been performed and passed without problem. The ISO document not only specifies different types of tests that must be performed for conformance testing, but also specifies a Test Plan (TP) architecture based on the ISO 9646-1 (ISO, 1994). The required TP is shown in figure 1. As can be seen from this figure, the TP architecture indicates that the tester should be divided into two parts. The first component is the Lower Tester (LT) which provides the test pattern generation and analysis. The second is termed the Upper Tester (UT), which is required to contain the software to control the CAN Implementation Under Test (IUT). The UT is normally a host processor or programmable device of some kind, and also provides coordination to conduct the tests between the LT and the IUT (Carmes et al, 1996). The UT receives stimulus (with details of the test being performed) from the LT, and generates messages passed on to the IUT. The IUT then processes these messages, and both the UT and LT components monitor its behaviour for consistency with the CAN protocol. If the result is satisfactory, the test is

considered passed and testing proceeds to the next conformance test. It should be noted that the testing procedures that are required to be implemented include coverage of common error conditions, randomized tests and also bit timing tests. Most tests are critical, and the latter category – bit timing – contains a number of tests that can be difficult to localize, and a suitable means is required to capture and display multiple logic signals over an appropriate timescale. This typically requires the use of dedicated hardware and Logic Analyzers (Lawrenz, 1998a).

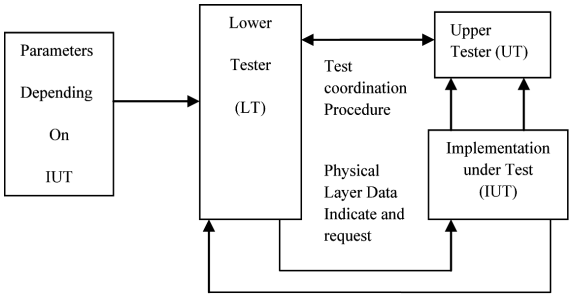


Figure 1: Conformance TP.

The motivation for the current work is as follows. Recent years have seen an increased interest in the employment of CAN-enabled devices implemented by programmable hardware devices such as FPGA's. By their very nature, such 'soft core' implementations are often needed in small-volume (or even one-off) batches. In these circumstances, cost and availability reasons often dictate that it is not practical for developers to use traditional CAN-conformance testing equipment. To help alleviate this problem, this paper proposes a low-cost and easily implemented method which will allow developers to test a CAN soft core implementation for conformance to the relevant standard without the need for expensive or proprietary hardware interfaces and logic analyzers.

The remainder of the paper is organized as follows. Section 2 discusses several approaches to CAN conformance testing that have been previously described. Section 3 describes the proposed test bed, and Section 4 presents two case studies that illustrate its use. Section 5 presents a comparison of the proposed approach to several other techniques, whilst Section 6 presents our initial conclusions.

2 PREVIOUS WORK

One of the earliest CAN prototype controllers was named DBCAN (Kirschbaum, 1996). This implementation was tested using a logic analyzer and a pattern generator circuit. As there was no standard for conformance testing at the time the prototype was developed, a commercial basic (as opposed to full) CAN controller was used as benchmark for verification. A major disadvantage of this scheme was the use of external interface modules to visualize the state of different DBCAN registers, and the testing procedure was somewhat limited in the number of signal channels that could be simultaneously analyzed. Since this is a needed requirement in the case of ISO standard conformance testing – the ability to visualize the state of large numbers of CAN registers simultaneously is a prerequisite – such a setup is limited in this respect.

A slightly different verification technique was reported by (Nimsub et al, 2005). Their technique employed custom design boards with 8051 microcontrollers and SJA1000 CAN controllers, but this method involved the design of specialized interface hardware and boards to assist with the testing plan. Specialised verification architecture for testing automotive protocols (including CAN) at both the module and chip level was proposed by (Zarri et al. 2006). Again, this work requires a specially designed CAN verification component as part of the silicon, while the selection and implementation of actual test sequences, along with the selection of a suitable means to monitoring bus signals, is left open for the tester.

With respect to soft core CAN implementations, the CAN e-Verification (CANeVC) test bench has previously been described (CANeVC 2005). This commercial test facility requires a CAN specification core to be embedded in the netlist; this core then runs specific tests to verify the behaviour of the CAN soft core. Again, this technique involves a time consuming development of a test bench using an expensive commercially available verification IP ; additionally, compatibility issues often arise when using CAN implementations other than the proprietary implementation (DiBlasi, 2003), and only a limited number of programmable logic devices are supported. Finally, several experimental implementations (such as that reported by (Ferreira et al. 2005) to measure single parameters - such as CAN bit errors - rather than perform complete conformance testing have been described in the literature. Such implementations have typically used complex and non-trivial means, requiring customized hardware and software. In summary then, it can be observed that - to date – specialised

hardware and / or software has been required to assist with CAN testing plans. In the following Section, a novel testing approach that relies only upon the use of low-cost, standard off-the-shelf hardware and software is described.

3 PROPOSED TEST BED

An ongoing project required the development of a conformant soft core controller for the CAN protocol (Sheikh et al. 2008). After the Verilog (IEEE, 2001) implementation of the CAN specifications for such a controller had been completed, it was required to be tested and verified in accordance with the relevant ISO standard (ISO 16845, 2000). Real-time testing of a CAN implementation is quite a complicated procedure, and – in this case – for practical reasons, no specialized hardware and software was available to generate the required testing patterns and monitor the behavior of the CAN soft core. For this reason, it was decided to use only low-cost off the shelf components.

In addition to these standard hardware parts, the Chipscope analysis tool (Xlinx, 2000) was used to visualize and capture the behavior of the soft core, allowing verification of the testing results. Chipscope is a Xilinx testing tool which is implemented the by inserting a small core onto the device to be monitored, allowing multiple signal channels to be captured via a JTAG interface. Up to 16 internal signal ports can be analyzed in a single core, and each port can have up to 256 signals. Multiple cores can be attached in a FPGA to increase the number of signals (Oltu et al. 2005). In comparison to other means for capturing multiple FPGA signals, Chipscope retains the key features required but is a fraction of the cost. Additionally, to support one-off conformance testing plans without causing excessive costs, a fully-featured evaluation version is available for a 60 day period – a full testing plan can be performed in such a timeframe. Hence these features of Chipscope made it an obvious choice for our CAN conformance test bed. The new test facility is shown schematically in figure 2. In the next paragraphs we give a full list of the hardware and software components and tools used in building the Test Bed.

3.1 Hardware

1) Two FPGA (XC3S500E programmed with CAN soft core) + ARM7 (LPC2138 as Host controller) boards. These boards are named as SC1 and SC2.
2) Two ARM7 Microcontroller boards with Integrated CAN controller and CAN transceiver for CAN bus interfacing named as KE and OL.
3) The SC1 and SC2 are connected to the bus using PCA82C250 CAN transceivers.
4) Parallel JTAG cable for downloading and analyzing signals for FPGA.
5) USB JTAG cable for downloading and debugging the ARM7 Microcontroller boards.

3.2 Software

1) Xilinx ISE for soft-core programming, synthesis, routing and programming the FPGA.
2) Chipscope Pro (Xilinx, 2000) is used as analysis tool (60 day evaluation version available).
3) The Keil uVision 3 IDE with free ARM tools C compiler was chosen for programming and debugging the Microcontroller boards.

As can be seen, the test bed has been made using COTS hardware and also taking in care the structure

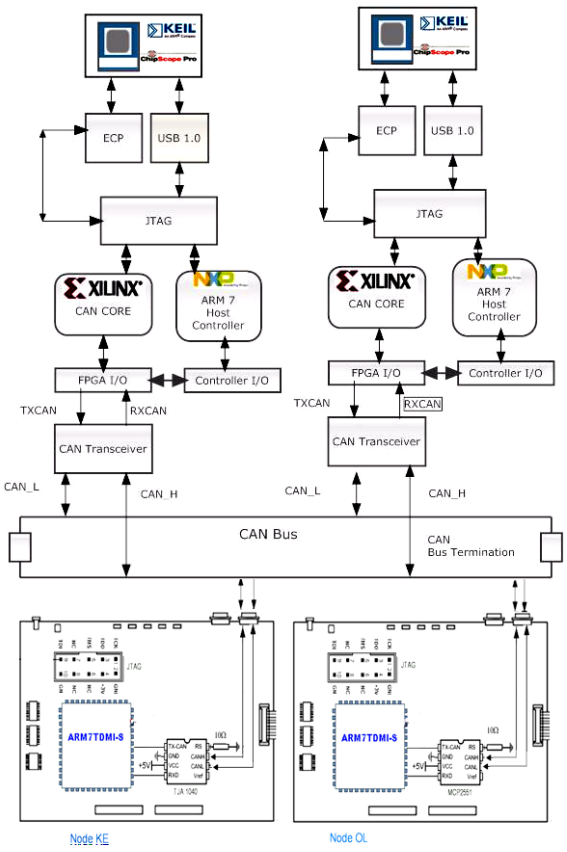


Figure 2: CAN Conformance test bed.

of the Test Plan given in ISO 9646-1. The test bed consists of two instances of the IUT; the main purpose of using the second IUT is to generate errors on the CAN bus and special conditions which were either the pre-requisite for a test case or generating special bit stream during a test for verifying the behavior of the IUT, hence the second instance of IUT is moreover working as the LT in reference to the ISO9646-1 TP. The ARM boards with integrated CAN Controllers were used either as receivers / transmitters to verify the conformance of the IUT with widely used CAN controllers, and were also employed to generate bit errors on the CAN bus using an interrupt generation mechanism. Such a scheme is highly synchronized as the bit inversions were done at the specific point where it was required; the methodology employed for test pattern generation is described in the next Section.

3.3 Test Pattern Generation

When using pattern generators test vectors are required to be first stored, and are sent on the CAN bus only when required – thus putting the IUT in different states and allowing its behaviour and responses to be analyzed. In our proposed test bed we have used FPGA based pattern generation, which is not only economical as no extra price was added to the test setup but also it is added as a Verilog module to the main CAN Core (IUT SC2 in Figure 2). This helped us to accurately produce these special conditions; for example in test case 1 (to be reported in the next Section) it was needed to produce extra dominant bits on the CAN bus after an IUT working as a transmitter send an Error Frame (ISO 16845, 2000). This test pattern was easily achieved by modifying the Verilog module for Error Flag generation to produce extra dominant bits, as illustrated by the Verilog code fragment shown opposite.

This is a simple example of pattern generation using HDL code. All of the required test patterns may be generated in this way, giving full controllability on the test case generation. In addition, tight synchronization of events can be achieved – a hardware signal from the core to one of the secondary ARM boards, at a certain point during the transmission of a CAN message, can be sent. This signal may be used to generate an interrupt on the ARM board – the interrupt latency is significantly less than a CAN bit-time, even at 1 Mbits/s – and within this ISR the ARM board can, for example, inject a bit error, The following Section describes two case studies to further highlight the operation of the test bed.

```
reg     [3:0]
Error_Flag_Counter;//changed from reg
[2:0]
always @ (posedge Clock or posedge
Reset)
begin
  if (rst)
     Error_Flag_Counter <= 4'd0;
  else if (Error_Frame_End |
Error_frame_Start)
      Error_Flag_Counter <=#delay 4'd0;
//changed from 3'd7
  else if (Error_Frame &
Transmit_Instance &Error_Flag_Counter <
4'd11))

Error_Flag_Counter <=#delay
Error_Flag_Counter + 1'b1;
end
always @ (Error_frame or
Error_Flag_Counter )
    begin
      if (Error_frame)         begin
if (Error_Flag_Counter < 4'd11)
//changed from    3'd7
            begin
              if (Node_Error_Passive)
                Tx_CAN = 1'b1;
              else
                Tx_CAN = 1'b0;
            End
```

4 CASE STUDIES

The proposed test facility was employed to test the CAN conformance of the custom created CAN soft core, written in Verilog. As the number of total number of test cases to consider in any single CAN conformance test plan is numerous, it is beyond the scope of the current paper to present comprehensive test results; such test results are available in the form of technical report (Sheikh & Short, 2009). However, in this Section we will present two test cases that help highlight the main features of the proposed facility. Both tests were carried out successfully, and are described in the following two Sections.

4.1 Error Flag Longer than 7 Bits

This test is a part of the Error Frame Management class in ISO 16845. The purpose of this test is to verify that a CAN transmitter will only tolerate 7 dominant bits after sending its own Error flag. The

case described below is for when the Error Flag is elongated by 4 Dominant bits. This test involves two instances of the IUT and the ARM7 Microcontroller boards. The test will be setup using the following organization:

1. Both the IUT's must be in default state ready for transmission or reception.
2. An error bit is to be introduced on the CAN bus during an ongoing transmission.
3. The transmitter - after sensing the error - must send an error frame of 6 dominant bits due to its Active Error state.
4. One of the receivers must send more than 7 dominant bits after receiving the Error flag.
5. The transmitter must not take these extra dominant bits as an Error and shouldn't send any extra Error Frame, and should start to resent the corrupted message.

The methodology employed was to modify one of the soft core IUTs to carry out this requirement. Any of the two IUT's can take the role of transmitter or receiver for any given test. In this case, the IUT instance which will be acting as a receiver is modified to generate an 11 bit Error Flag. The snapshot of the events on the CAN bus was captured with the help of Chipscope trigger mechanism (Woodward, 2003). The observations on the transmitter node from the Chipscope snapshot – shown in figure 3 - are as follows:

1. On the Left of Marker 'T', the TX_state high indicates an ongoing transmission. During data transmission a Bit_Error is injected, indicated by the bit inversion as CAN_Tx is recessive while CAN_Rx is dominant (Bit inversion is done by the KE node).
2. The Error Frame exists between markers 'T' and 'O' indicated by signal 'Error_Frame', the Error_Flag is between Marker 'T' and 'X'. The end of Error Flag is indicated by a high Error_Flag_Tx_over signal. The Error_Flag_Counter three bit bus is indicating the count of Error Flag bits sent.
3. If we note at Marker 'X' the CAN_Tx signal has changed to Recessive but the CAN_Rx signal remains Dominant for next 4 bits which is because of the superimposition of the Error Flag sent by the nodes on the CAN network.
4. The Error Frame is continued till Marker 'O' and the end is shown by a low Error_Frame Signal and a high Error_Frame_End Signal.
5. On the right of Marker 'O' we can clearly see that after three sample points (intermission Field) a new frame transmission has started indicated by Tx_State, while Tx_State_ID[10:0] and Tx_State_Data indicating different states of transmission cycle.

The observations at the receiver node (shown in figure 4) are as follows:

1. Marker 'O' indicates start of an Error_Frame, the Tx_State low indicating the node is a receiver.
2. The Error_Flag_Counter is a 4 bit wide bus which counts up to 11 bits i.e. it is sending 4 extra Dominant bits to the CAN bus to verify the behaviour of transmitter node. The Dominant bits can also be verified by the CAN_Tx and CAN_Rx bits.
3. After the Error_Flag_Over Signal is set high the CAN_Rx and CAN_Tx signals turns to dominant for next seven bits indicating an Error frame Delimiter.
4. Right of Marker 'X' the Error_Frame signal is low and after three Recessive bits (Intermission Field) a new frame is started to be received (Tx_State is low, CAN_Tx is recessive), indicated by different Recieve_State_xxxx signals).

4.2 Overload Frame Management

This test is a part of the Overload Frame Management class (ISO, 2000). This test verifies that an IUT will be able to transmit a data frame starting with the identifier and without transmitting SOF, when detecting a dominant bit on the third bit of the intermission field. This test involves two instances of the IUT and the ARM7 Microcontroller boards. The test will be setup using the following organization:

1. Both of the IUT's must be in default state ready for transmission or reception according to the setup sent by the Host Controller.
   The IUT acting as the Transmitter is set to transmit two data frames as programmed in the Host processor.
2. The Receiver IUT will be set to request an Overload frame after reception of the first frame.
3. After the completion of the Overload Frame on the third bit of the Intermission field (Normally the Intermission field is a sequence of three Recessive bits) is set to dominant by the Fault injector node i.e. K.E.
4. The transmitter must not consider it as a bit error and shouldn't send a Dominant level SOF and consider the dominant bit of the Intermission field as the SOF.

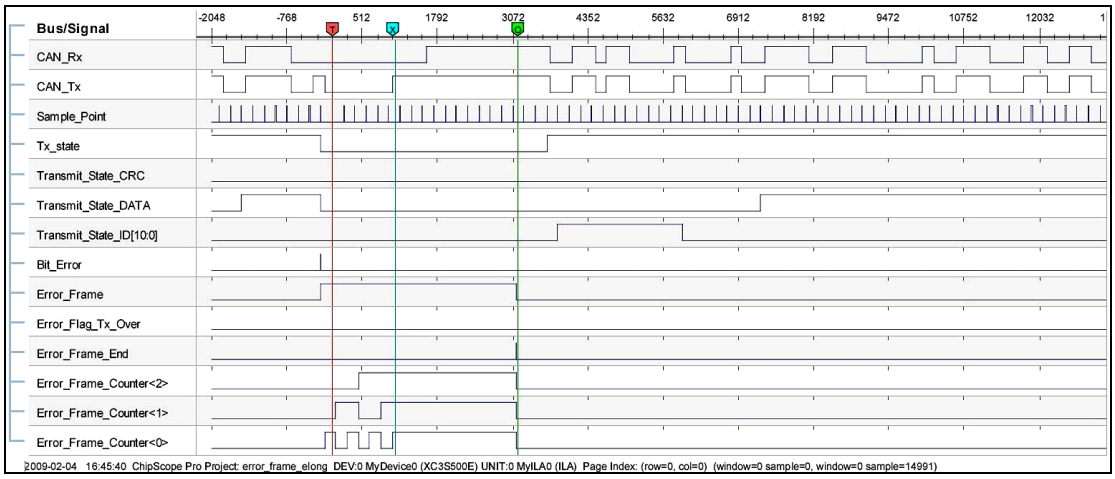


Figure 3: Transmitter snapshot for Test Case 1.

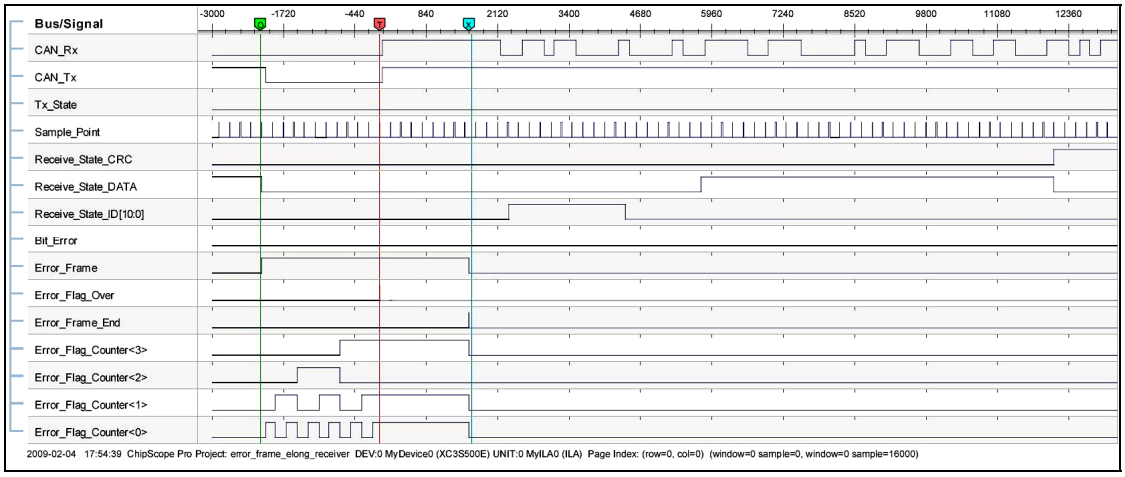


Figure 4: Receiver snapshot for Test Case 1.

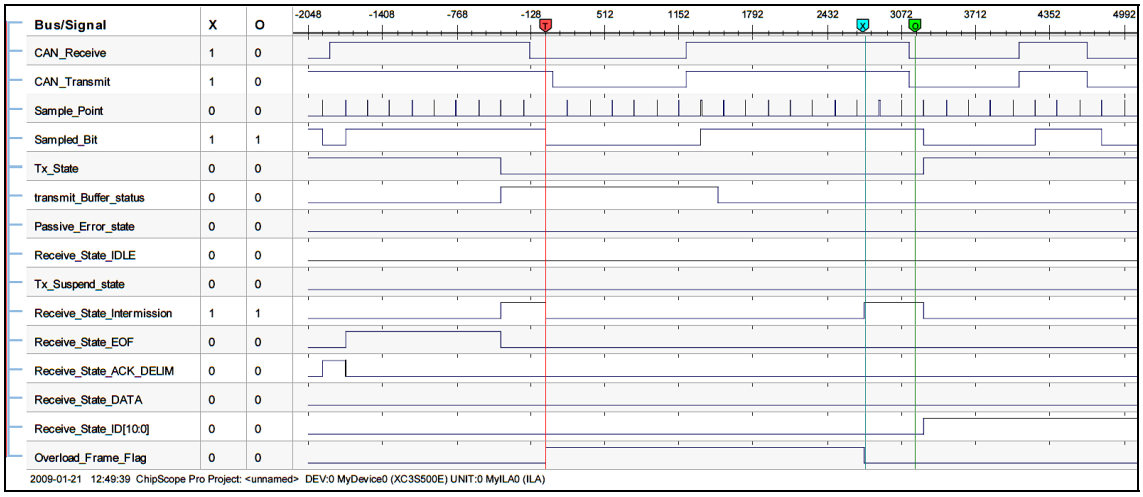


Figure 5: Transmitter snapshot for Test Case 2.

5. Normal reception of the message should take place.

This test was successful with desired results as stated in the purpose of the test; the observation on the transmitter node from the Chipscope - shown in figure 5 – is as follows:

1. Left of Marker 'T' The Tx_state flag is high indicating ongoing transmission, Receive_state_Data and ACK_DELIM indicating a successful transmission while the node is error active.
2. At Marker 'T' there is an error on the Receive_state_intermission field generating an overload frame with Overload Flag of six dominant and Overload delimiter of 8 bits as can be seen by the count of sample point.
3. After the Overload frame an intermission field signal can be seen at the Marker 'X'.
4. The third bit of intermission field is is a dominant bit as can be counted between Markers 'X' and 'O' the number of sample points is 2 and the third sample point is a dominant bit.
5. Just after the Marker 'O' we can see the Receive_State_ID [10:0] going high without any SOF. The Identifier first 4 bits are dominant as required by the Test case.

5 COMPARATIVE STUDY

When comparing the proposed testbed with previous methods, the first observation is that the current facility does not require the use of expensive CAN PC interface cards which are normally required for CAN conformance testing (Lawrenz, 1998b). Such cards also require specialized software (along hardware and interface cables) which can adds to the cost and complexity of the setup. In the proposed implementation the internal state of CAN IUT is analyzed directly using Chipscope, and also by using the Keil uVision to debug on the ARM boards. In addition, there are several key advantages of our proposed test bed using Chipscope over hardware logic analyser systems:

1. The standard bench analyzers doesn't show enough signals as required in case of CAN conformance as illustrated in section 4. There are Logic analyzer systems which can show large number of signals simultaneously with large data widths but there prices are 10 times more than Integrated Logic analyzer.
2. Normal Bench analysers can show Mega samples, while the Chipscope is limited to a Sample width of 16K, we overcome this problem by using Digital clock Manager which can divide or multiply the system clock by 'n' times, the board we used in our system can divide the system clock by 16 times hence we were able to capture 16 times more sample than on system clock which can easily capture 3 to 4 complete CAN messages in a single trigger.
3. Additional probes with wide numbers of I/O pins are required to interface with the Logic analysers while Chipscope can carry magnitude of these signals using a simple JTAG cable, although there are few solution like Agilent's FPGA trace port (Agilent, 2003) which use a simple interface to analyse multiple signals but it also requires a specialized hardware and Chipscope pro tool.
4. Not only all I/O signals are accessible through Chipscope but also internal wires can be traced (Lee, 2007) which are really helpful in Conformance testing specially when setting up triggering conditions we have lot more options to setup a trigger condition for example in the test cases discussed it is really easy to setup a trigger condition to wait for an Error Frame flag signal goes to high to analyze an error condition, while for external Logic analyzers only I/O signals are available.

6 CONCLUSIONS

In this paper we have presented a low cost and flexible approach to CAN conformance testing in accordance with the ISO standards. It has been shown that the facility is capable of performing the full range of test required to show conformance to the relevant CAN standard (Bosch, 1991), (ISO, 2003). In conclusion, this facility can be assembled and used for a fraction of the cost of a 'regular' test facility for CAN conformance. A full list of the how each individual test may be implemented when using a facility such as this has been described in a technical report (Sheik & Short, 2009), allowing the facility to be implemented by any third-parties requiring a low-cost methodology to test for conformance of a CAN soft-core.

As a final note, it can be seen that test facility that has been described is not restricted to the CAN protocol, and – with suitable modifications – can be used to test conformance of many alternate network protocols, for example TTP/C (TTA, 2003).

ACKNOWLEDGEMENTS

This Paper is part of Imran Sheikh's PhD studies which is financed by NWFP University of Engineering & Technology Peshawar, Pakistan.

REFERENCES

Agilent Technologies, 2003. Deep Storage with Xilinx Chipscope Pro and Agilent Technologies FPGA Trace Port Analyzer. http://cp.literature.agilent.com/litweb/pdf/5988-7352EN.pdf

Bosch, R., 1991. CAN Specification 2.0, Postfach, Stuttgart, Germany: Robert Bosch GmbH.

CAN 2.0 eVC, 2005. Yogitech SPA.

Carmes, E., Junier, C., and Aussedat, F., 1996. CAN Conformance: Methodology and Tools, Keynote speech, CAN in Automation Proceedings of 3rd iCC 1996, Paris, October 1996

Di Blasi, A., Colucci, F., and Mariani, R., 2003. Y-CAN Platform: A Re-usable Platform for Design, Verification and Validation of CAN-Based Systems On a Chip, ETS- 2003 Symposium, May2003

Ferreira, J., Oliveira, A., and Fonesca, J., 2005. An Experiment to Assess Bit Error Rate in CAN, In Proceedings of 3rd International Workshop of Real-time Networks (RTN 2004), Catania, Italy.

IEEE, 2001. Standard for Verilog Hardware Description Language, IEEE standard 1364.

ISO, 1994. DIS 9646-1 The International Organization for Standardization. Information technology-Open Systems Interconnection-Conformance testing methodology and framework.

ISO, 2000. DIS-16845, Road Vehicles- Controller Area Network (CAN) - Conformance Test Plan

ISO, 2003. DIS 11898-1 Road vehicles – Controller area network (CAN) – Part 1: Controller area network data link layer and physical signalling.

Kirschbaum, A.; Renner, F.M.; Wilmes, A.; Glesner, M., 1996. Rapid-prototyping of a CAN-Bus controller: a case study, Rapid System Prototyping, 1996. Proceedings. Seventh IEEE International Workshop on , vol., no., pp.146-151, 19-21 Jun 1996.

Lawrenz, W., Kinowski, P., and Kircher,G., 1998a. CAN Conformance Testing-The Developing ISO Standard and Necessary Extensions, In Proceedings of International Truck and Bus Meeting and Exposition Indianapolis, Indiana, November 16-18, 1998.

Lawrenz, W., Kinowski, P. and Kircher, G., 1998b. CAN Conformance Testing - State of the Art and Test Experience, In Proceedings of 5th International CAN Conference iCC'98, San Jose, California, November 1998.

Lee, T., Fan, Y., Yen, S., Tsai, C., and Hsiao, R., 2007. An Integrated Functional Verification Tool for FPGA Systems, Second International Conference on Innovative Computing, Information and Control, ICICIC '07, pp.203-203 5-7 Sept. 2007.

Nimsub, K., Dawi, K., Kyuhyung, C., Jinsang, K., and Wonkyung, C., 2005. Design and Verification of a CAN Controller for Custom ASIC, CAN in Automation Proceedings of 10th iCC 2005.

Oltu, O., Milea, P., Simion, A., 2005. Testing of digital circuitry using Xilinx Chipscope logic analyzer, In Proceedings International Semiconductor Conference, CAS 2005, vol.2, no., pp. 471-474, 3-5 Oct. 2005.

Sheikh, I., Short. M, and Pont, M., 2008. Hardware Implementation of a Shared Clock Protocol for CAN: A Pilot Study, In proceedings of 4th UK Embedded Forum, Southampton, September, 2008.

Sheikh, I., and Short, M., 2009. CAN Conformance Testing-A New approach, tech-report ESL-09-01, ESL, Engineering Department, University of Leicester.

TTA-Group, 2003. Time-Triggered Protocol TTP/C High-Level specification Doc. Protocol Ver. 1.1, 1.4.3 ed. Vienna, Austria, TTTECH.

Xilinx Inc, 2000. Chipscope integrated logic analyzer, San Jose, CA 95124-3400. http://www.xilinx.com/ise/optional_prod/cspro.htm

Woodward, J., 2003. The in-circuit debug of FPGAs, CMP Media LLC, New York, Embedded Systems Europe, vol 7, No 49, pp.16-17.

Zarri, G., Colucci, F., Dupuis, F., Mariani, R., Pasquariello, M., Risaliti, G. and Tibaldi,C., 2006. On the verification of automotive protocols, In Proceedings of Design, Automation and Test in Europe, 2006. DATE '06. , vol.2, no., March 2006, pp.6-10.

FUZZY LOGIC BASED QUADROTOR FLIGHT CONTROLLER

Syed Ali Raza and Wail Gueaieb
School of Information Technology and Engineering, University of Ottawa
800 King Edward Avenue, Ottawa, ON, Canada
syedali.raza@uottawa.ca, wgueaieb@site.uottawa.ca

Keywords: Quadrotor, Fuzzy Logic, Flight Controller.

Abstract: Quadrotor unmanned aerial vehicles (UAVs) have gained a lot of research interest in the past few years, due to the clear advantages posed by their vertical take-off and landing (VTOL), hovering capability, and slow precise movements. These characteristics make quadrotors an ideal candidate for applications that require traversing through difficult environments with many obstacles. Belonging to the helicopter rotorcraft class, quadrotors are highly nonlinear systems that are difficult to stabilize. This paper proposes a fuzzy logic based flight controller for an autonomous quadrotor. Two types of fuzzy logic controllers are implemented. The developed flight controllers are tested in a quadrotor simulator and simulation results are presented to demonstrate the performance of each controller. The controllers performances are also benchmarked against conventional control based techniques such as input-output linearization, backstepping and sliding mode control. In comparison with other conventional control techniques mostly designed for indoor applications, the proposed fuzzy logic based controllers showed satisfactory control of the quadrotor in the presence of various disturbances such as sensor noise and high wind conditions.

NOMENCLATURE

The following notations are used in the paper:
$\mathcal{F}_n$ represents the reference frame where the subscript $n \in \{i, b, v, \phi, \theta\}$.
$()_{\mathcal{F}_n}$ represents a point or a vector in reference frame $\mathcal{F}_n$.
$\dot{()}, \ddot{()}$ represent first and second time derivatives, respectively.
$R_{\mathcal{F}_1}^{\mathcal{F}_2} \in \mathbb{R}^{n \times n}$ is the rotation matrix that maps frame $\mathcal{F}_1$ to frame $\mathcal{F}_2$.
$s\theta = \sin\theta$, $c\theta = \cos\theta$.
I is the identity matrix.
COG is the quadrotor's center of gravity.
$P^T = [p_x \ p_y \ p_z]$ is the position of the quadrotor's COG.
M is the mass of quadrotor (including the motors).
m is the mass of one motor.
l is the length of the arms of quadrotor.
$J^T = [j_x \ j_y \ j_z]$ is the moment of inertia vector of the quadrotor.
ϕ, θ, ψ are the quadrotor's roll, pitch, and yaw angles, respectively.
$\Omega^T = [\phi \ \theta \ \psi]$ is the quadrotor's orientation vector.
K_T and K_τ are motor constants.
$T_{motor}, \tau_{motor}, \text{PWM}_{motor}$ are thrust, drag and PWM value for $motor \in \{f, r, b, l\}$, the subscripts f, r, b, and l, denote front, right, back, and left, motors respectively.

1 INTRODUCTION

The motivation for employing robots in search and rescue operations is multifaceted. However, the technology is still in its infancy and there are many issues which need to be addressed. Search and rescue missions, as well as simulations, have demonstrated several areas in which robot contributions need to be improved (Fincannon et al., 2004). Advances in computing, MEMS inertial measurement sensors, and communications technology make it possible to achieve autonomous performance and coordination of these vehicles in test environments (Waslander et al., 2005). But, achieving complete autonomous performance in real environments is a milestone yet to be reached.

A quadrotor, as depicted in Figure 1, is a rotary wing UAV, consisting of four rotors located at the ends of a cross structure. By varying the speeds of each rotor, the flight of the quadrotor is controlled. Quadrotor vehicles possess certain essential characteristics, which highlight their potential for use in search and rescue applications. Characteristics that

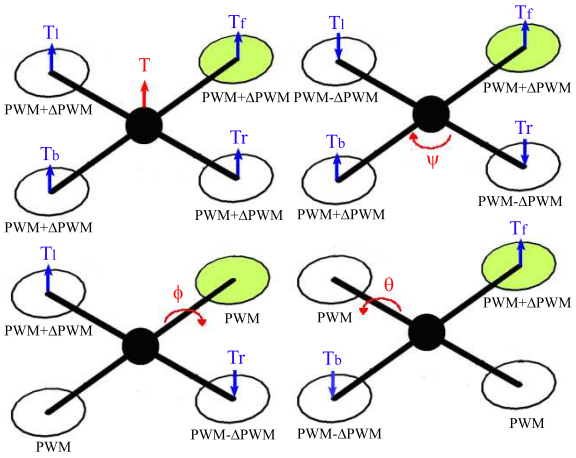


Figure 1: Conceptual diagram of the quadrotor.

provide a clear advantage over other flying UAVs include their Vertical Take Off and Landing (VTOL) and hovering capability, as well as their ability to make slow precise movements. There are also definite advantages to having a four rotor based propulsion system, such as a higher payload capacity, and impressive maneuverability, particularly in traversing through an environment with many obstacles, or landing in small areas.

The quadrotor is an under-actuated system, in that the four forces applied on the body result in movements achieved in six degrees. Being a non-linear system it offers an interesting and challenging control problem. The flight controller is responsible for achieving the desired attitude, i.e. the orientation (roll, pitch and yaw angles) of the quadrotor and also the desired position of quadrotor in the space.

In past few years, a lot of research has already been conducted on the modeling and control of a quadrotor. Many control techniques are proposed in the literature, however, their primary focus is mostly for indoor flight control and therefore does not account for uncertainties. Lyapunov stability theory is used for stabilization and control of the quadrotor in (Bouabdallah et al., 2004a) (Dzul et al., 2004). Conventional PD^2 feedback, and PID structures are used for simpler implementation of control laws and comparison with LQR based optimal control theory is presented in (Tayebi and McGilvray, 2006) (Bouabdallah et al., 2004b). Backstepping control is also proposed with the drawback of higher computational loads in (Guenard et al., 2005). Visual feedback is used in many cases using onboard or offboard cameras for pose estimation by (Altug et al., 2002) (Guenard et al., 2008). And finally some fuzzy logic based control techniques (Coza and Macnab, 2006), neural networks (Tarbouchi et al., 2004) and reinforcement learning (Waslander et al., 2005).

Fuzzy logic based control offers a great advantage over conventional control, specifically in dealing with nonlinear systems with uncertainties. In this paper two fuzzy logic based flight controllers for an autonomous quadrotor are proposed. Implementation of the proposed control is carried out using Mamdani and Sugeno fuzzy inferencing methodologies. The advantage of the proposed methodology over other conventional methods is the ability to achieve stable flight control under disturbances that are common in outdoor environments. Also the implementation is lighter in terms of computational loads as compared to techniques such as backstepping control. Both controllers are simulated and their performances are benchmarked against various conventional control techniques. The results are presented to show the satisfactory performance of the proposed fuzzy logic based controllers despite the presence of various disturbances, such as sensor noise and high wind conditions.

2 QUADROTOR'S KINEMATICS AND DYNAMICS

The flight behavior of a quadrotor is determined by the speeds of each of the four motors, as they vary in concert, or in opposition with each other. Hence, a mathematical representation of the system can be used to predict the position and orientation of the quadrotor, based on its inputs. The same can further be used to develop a control strategy, whereby manipulating the speeds of individual motors results in the achievement of the desired motion. To derive a mathematical model of the quadrotor, we need to define its kinematics and dynamics. The kinematics of the quadrotor gives a relation between the position of the vehicle in the inertial frame, and its velocity in the body frame. The dynamics of the quadrotor provides the relation between the applied body forces and the resulting accelerations.

Three frames of reference are adopted;

1. The inertial frame, $\mathcal{F}_i = (\overrightarrow{x_i}, \overrightarrow{y_i}, \overrightarrow{z_i})$, is an earth-fixed coordinate system with the origin located on the ground, for example, at the base station. By convention, the x-axis points towards the north, the y-axis points towards the east, and the z-axis points towards the center of the earth.
2. The body frame, $\mathcal{F}_b = (\overrightarrow{x_b}, \overrightarrow{y_b}, \overrightarrow{z_b})$, with its origin located at the center of gravity (COG) of the quadrotor, and its axes aligned with the quadrotor structure such that the x-axis $\overrightarrow{x_b}$ is along the arm

with front motor, the y-axis $\overrightarrow{y_b}$ is along the arm with right motor, and the z-axis $\overrightarrow{z_b} = \overrightarrow{x_b} \times \overrightarrow{y_b}$.

3. The vehicle frame, $\mathcal{F}_v = (\overrightarrow{x_v}, \overrightarrow{y_v}, \overrightarrow{z_v})$, is the inertial frame with the origin located at the COG of the quadrotor. The vehicle frame has two variations, $\mathcal{F}_\phi$ and $\mathcal{F}_\theta$. $\mathcal{F}_\phi$, is the vehicle frame, $\mathcal{F}_v$, rotated about its z-axis $\overrightarrow{z_v}$ by an angle ψ so that $\overrightarrow{x_v}$ and $\overrightarrow{y_v}$ are aligned with $\overrightarrow{x_b}$ and $\overrightarrow{y_b}$, respectively. $\mathcal{F}_\theta$ is the $\mathcal{F}_\phi$ frame rotated about its y-axis, $\overrightarrow{y_\phi}$, by a pitching angle, θ, such that $\overrightarrow{x_\phi}$ and $\overrightarrow{z_\phi}$ are aligned with $\overrightarrow{x_b}$ and $\overrightarrow{z_b}$, respectively.

With the knowledge of the inertial frame position state variables P_i and the body frame speed state variables $\dot{P}_{\mathcal{F}_b}$, the translational motion relationship is derived as

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ -\dot{p}_z \end{bmatrix}_{\mathcal{F}_i} = \left[R^{\mathcal{F}_b}_{\mathcal{F}_v}\right]^T \begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{p}_z \end{bmatrix}_{\mathcal{F}_b}$$

where $\left[R^{\mathcal{F}_b}_{\mathcal{F}_v}\right]^T \in \mathbb{R}^{3\times3}$ is the rotation that maps frame $\mathcal{F}_b$ to frame $\mathcal{F}_v$ and is defines by

$$\left[R^{\mathcal{F}_b}_{\mathcal{F}_v}\right]^T = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix}$$

The rotational motion relationship is also derived using the state variables involved:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{\mathcal{F}_v} = \begin{bmatrix} 1 & s\phi\tan\theta & c\phi\tan\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{\mathcal{F}_b}.$$

The quadrotor's dynamics is obtained by applying Newton-Euler formulation.

$$\begin{bmatrix} MI_{3\times3} & 0 \\ 0 & I_{3\times3} \end{bmatrix} \begin{bmatrix} \ddot{P}_{\mathcal{F}_b} \\ \dot{\Omega}_{\mathcal{F}_b} \end{bmatrix} + \begin{bmatrix} \dot{\Omega}_{\mathcal{F}_b} M\dot{P}_{\mathcal{F}_b} \\ \dot{\Omega}_{\mathcal{F}_b} I\Omega_{\mathcal{F}_b} \end{bmatrix} = \begin{bmatrix} F_{\mathcal{F}_b} \\ \tau_{\mathcal{F}_b} \end{bmatrix}$$

where $F^T = [f_x\ f_y\ f_z]$ and $\tau^T = [\tau_\phi\ \tau_\theta\ \tau_\psi]$ are external force and torque vectors applied on the quadrotor's COG. τ_ϕ, τ_θ, and τ_ψ are the roll, pitch and yaw torques respectively. Hence, the models for the translational and rotational motion are

$$\begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \\ \ddot{p}_z \end{bmatrix}_{\mathcal{F}_b} = \begin{bmatrix} \dot{\psi}\dot{p}_y - \dot{\theta}\dot{p}_z \\ \dot{\phi}\dot{p}_z - \dot{\psi}\dot{p}_x \\ \dot{\theta}\dot{p}_x - \dot{\phi}\dot{p}_y \end{bmatrix}_{\mathcal{F}_b} + \frac{1}{M} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}_{\mathcal{F}_b}$$

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix}_{\mathcal{F}_b} = \begin{bmatrix} \frac{j_y - j_z}{j_x}\dot{\theta}\dot{\psi} \\ \frac{j_z - j_x}{j_y}\dot{\phi}\dot{\psi} \\ \frac{j_x - j_y}{j_z}\dot{\phi}\dot{\theta} \end{bmatrix}_{\mathcal{F}_b} + \begin{bmatrix} \frac{1}{j_x}\tau_\phi \\ \frac{1}{j_y}\tau_\theta \\ \frac{1}{j_z}\tau_\psi \end{bmatrix}_{\mathcal{F}_b}$$

where $j_x = j_y = j_z = \frac{2Mr^2}{5} + 2l^2m$. The moment of inertia is calculated by assuming the COG of the quadrotor as a sphere of radius r and mass M. Also, each motor is represented by a point mass m with arms length l. The relationship between the quadrotor's overall thrust and drag produced by individual thrust of the motors is defined by

$$T = T_f + T_r + T_b + T_l\ ,\ \tau_\psi = (\tau_f + \tau_b) - (\tau_r + \tau_l)$$
$$\tau_\phi = l(T_l - T_r)\ ,\ \tau_\theta = l(T_f - T_b)$$

with

$$T_{motor} = K_T \times \mathrm{PWM}_{motor} \text{ and } \tau_{motor} = K_\tau \times \mathrm{PWM}_{motor}$$

for $motor \in \{f, r, b, l\}$. In matrix form, we get

$$\begin{bmatrix} \mathrm{PWM}_f \\ \mathrm{PWM}_r \\ \mathrm{PWM}_b \\ \mathrm{PWM}_l \end{bmatrix} = G \times \begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix}$$

where

$$G = \begin{bmatrix} K_T & K_T & K_T & K_T \\ 0 & -l \times K_T & 0 & l \times K_T \\ l \times K_T & 0 & -l \times K_T & 0 \\ -K\tau & K\tau & -K\tau & K\tau \end{bmatrix}^{-1}.$$

Including the forces and torques acting on the system, the equations of motion become as defined in (1), with g being the gravitational force.

3 FLIGHT CONTROLLER DESIGN

Using the knowledge of the quadrotor's behavior from its equations of motion, a rule base is developed, which serves as a starting point for developing the fuzzy logic control strategy. These IF-THEN rules will later be combined to form a fuzzy controller, where fuzzy logic is used in a direct control scheme applied for the flight control of the quadrotor.

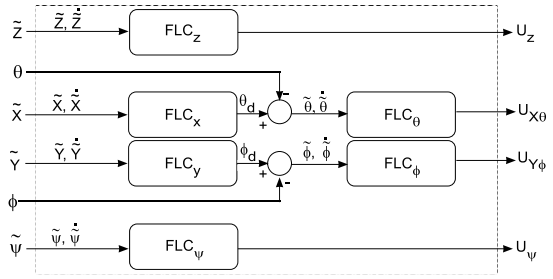


Figure 2: Block diagram of fuzzy logic based flight controller.

For attitude control, three fuzzy controllers are designed to control the quadrotor's roll (ϕ), pitch (θ)

$$\begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \\ \ddot{p}_z \end{bmatrix}_{\mathcal{F}_b} = \begin{bmatrix} \dot{\psi}\dot{p}_y - \dot{\theta}\dot{p}_z \\ \dot{\phi}\dot{p}_z - \dot{\psi}\dot{p}_x \\ \dot{\theta}\dot{p}_x - \dot{\phi}\dot{p}_y \end{bmatrix}_{\mathcal{F}_b} + \begin{bmatrix} -gs\theta \\ gc\theta s\phi \\ gc\theta c\phi \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{-f_z}{M} \end{bmatrix}, \quad \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix}_{\mathcal{F}_b} = \begin{bmatrix} \frac{j_y - j_z}{j_x}\dot{\theta}\dot{\psi} \\ \frac{j_z - j_x}{j_y}\dot{\phi}\dot{\psi} \\ \frac{j_x - j_y}{j_z}\dot{\phi}\dot{\theta} \end{bmatrix}_{\mathcal{F}_b} + \begin{bmatrix} \frac{1}{j_x}\tau_\phi \\ \frac{1}{j_y}\tau_\theta \\ \frac{1}{j_z}\tau_\psi \end{bmatrix}_{\mathcal{F}_b} \quad (1)$$

and yaw (ψ) angles. Here FLC_ϕ and FLC_θ fuzzy controllers are implemented in order to achieve attitude stabilization. Three fuzzy controllers are further designed for controlling the quadrotor's position. All the six fuzzy controllers, as depicted in Figure 2, have the same structure with two inputs and one output. The inputs are the error e, which is the difference between the desired and the actual state normalized to the interval $[-1,+1]$, and the error rate $\dot{e}$ normalized to the interval $[-3,+3]$. Three membership functions are used to fuzzify e, $\dot{e}$, and the output U: $\mu_N(e) = \text{trapezoid}(-1,-0.15,0)$, $\mu_Z(e) = \text{triangle}(-0.15,0,0.15)$, $\mu_P(e) = \text{trapezoid}(0,0.15,1)$, $\mu_N(\dot{e}) = \text{trapezoid}(-3,-1.5,0)$, $\mu_Z(\dot{e}) = \text{triangle}(-1.5,0,1.5)$, $\mu_P(\dot{e}) = \text{trapezoid}(0,1.5,3)$, $\mu_N(U) = \text{trapezoid}(-1,-0.85,0)$, $\mu_Z(U) = \text{triangle}(-0.1,0,0.1)$, $\mu_P(U) = \text{trapezoid}(0,0.85,1)$.

A unified rule base comprising nine IF-THEN rules is developed for all the controllers. The developed rules are presented in Table 1.

Table 1: The rule base of the fuzzy controller.

$\dot{e}$ \ e	N	Z	P
N	N	N	Z
Z	N	Z	P
P	Z	P	P

For the fuzzy controller to be independent of the quadrotor's parameters, the input and output signals of the designed fuzzy logic controllers are pre-processed and post-processed, respectively. The pre-processing process calculates the error e and error rate $\dot{e}$ and normalize them accordingly. The post-processing process calculates the individual motor PWM values by combining the outputs from the fuzzy logic controllers and adding a priori-defined offset to counter balance the weight of the quadrotor. The PWM value of each motor is then computed as follows:

$$\begin{aligned} \text{PWM}_f &= Sat(U_Z + U_{X\theta} - U_\psi + \text{Offset}) \\ \text{PWM}_r &= Sat(U_Z + U_{Y\phi} + U_\psi + \text{Offset}) \\ \text{PWM}_b &= Sat(U_Z - U_{X\theta} - U_\psi + \text{Offset}) \\ \text{PWM}_l &= Sat(U_Z - U_{Y\phi} + U_\psi + \text{Offset}) \end{aligned}$$

Being independent of the plant's parameters sets the pure fuzzy controllers apart from the conventional control systems, which depends on the plant's parameters. The fuzzy controllers are designed in light of the behaviors extracted from the mathematical model of the quadrotor, whereas conventional control techniques are designed using the mathematical model of the plant. Any change in the plant dynamics or parameters results in the failure of such conventional control systems. In such situations, fuzzy systems are advantageous over conventional control techniques.

Two different fuzzy inference procedures are implemented: (i) a Mamdani fuzzy model, and (ii) a Takagi-Sugeno-Kang (TSK) fuzzy model. The Mamdani fuzzy inference method uses a min-max operator for the aggregation and centroid of area method for defuzzification. One problem with the designed Mamdani fuzzy control is the high computational burden associated to it when implemented on an embedded system. To alleviate this problem, a zero order Sugeno fuzzy model is implemented. In this model, the output membership functions of the Mamdani type FLC are replaced with fuzzy singletons $N = -1$, $Z = 0$ and $P = +1$.

4 NUMERICAL RESULTS

To test the proposed fuzzy flight controllers, and study their performances, a simulation environment is developed. The desired inputs from the user are the translatory coordinates $P_{\mathcal{F}_i}$ with respect to the inertial frame, and the yaw angle ψ. The pitch and roll angles are set to zero for achieving the desired attitude stabilization, as shall be shown later.

To make the simulations more realistic, sensory noise and environmental disturbances such as winds are also taken into account. Different wind conditions are generated based on the actual data from Canada Weather Statistics, representing high, medium and low wind conditions (Statistics, 2009). The angular accelerations $\ddot{\phi}$, $\ddot{\theta}$, and $\ddot{\psi}$ are degraded with a white noise and then used as a feedback to the fuzzy controller.

Several experiments are conducted on the two designed fuzzy controllers. Due to the space limitation, only three experiments are presented here. The controllers performances are benchmarked against those of similar control schemes found in the literature. The values used as the quadrotor's dynamic

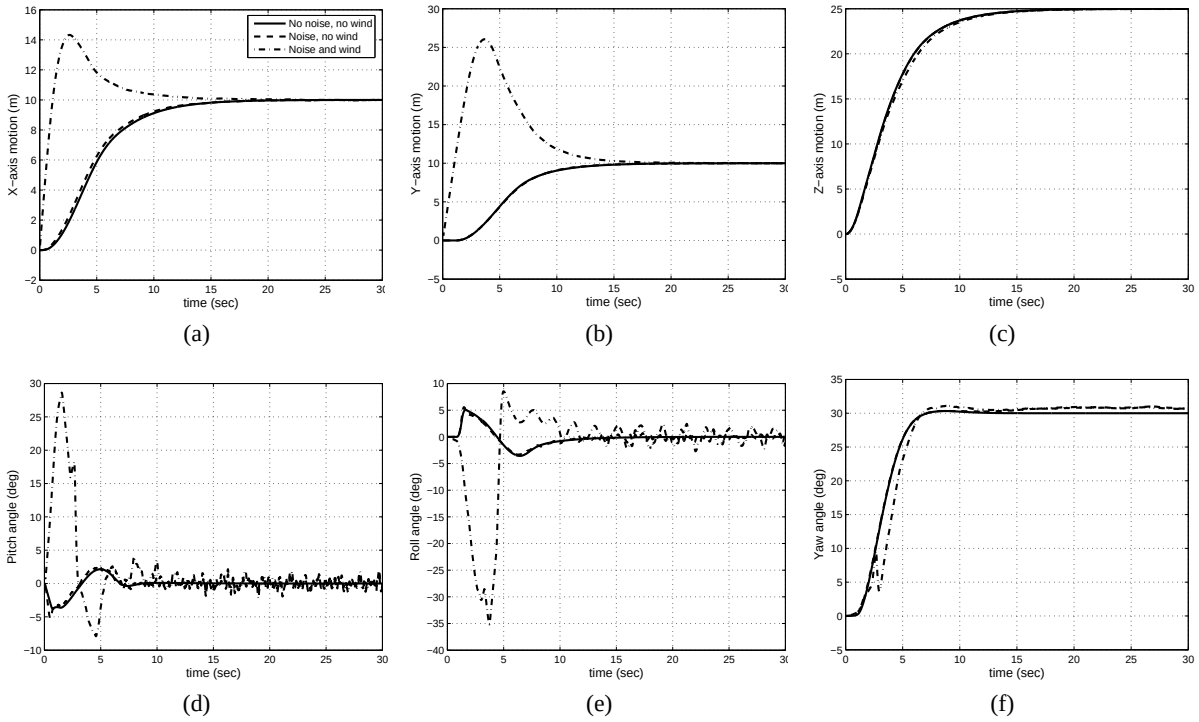


Figure 3: Experiment 1, Mamdani FLC's performance with and without disturbances. Quadrotor states: (a) x-axis; (b) y-axis; (c) z-axis (altitude); (d) pitch (θ); (e) roll (ϕ); and (f) yaw (ψ).

parameters are: $M = 1.56$ Kg, $l = 0.3$ m, $J_x = J_y = 0.08615$ Kg·m^2, $J_z = 0.1712$ Kg·m^2, $K_T = 5.45$, $K_{tau} = 0.0549$, Offset $= Mg/(4K_T) = 0.7018$.

In our first experiment the system's initial states are set to zero and the desired quadrotor's position and orientation are $P^T_{\mathcal{F}_i} = [10, 10, 25]$ m and $\Omega^T_{\mathcal{F}_i} = [0, 0, 30]$ degrees. The purpose of the experiment is to show the performance of both controllers under different disturbance conditions. The experiment is repeated three times for each controller, first time without any disturbances, second time with sensor noise only, and third time with sensor noise and medium north and east wind of $10m/s$. The experimental results presented in Figures 3 and 4, demonstrate the ability of both controllers to perform satisfactorily despite the presence of sensor noise and wind disturbances. The Mamdani fuzzy controller converges to the desired states relatively faster than the Sugeno fuzzy controller, however Sugeno shows better attitude stabilization, specially in the presence of the sensory noise and wind disturbances. The yaw angle drift under wind disturbance is clearly visible in the Sugeno results.

In second experiment, the position and attitude stabilization of the proposed controllers are benchmarked against feedback linearization based conventional control technique, where a series of mode-based, feedback linearizing controllers are implemented for the stabilization and control of a quadrotor (Altug et al., 2002). The system's initial states are considered to be $P^T_{\mathcal{F}_i} = [40, 20, 60]$ m and $\Omega^T_{\mathcal{F}_i} = [45, -45, 0]$ degrees. In the presence of sensor noise and winds, the system is supposed to stabilize all the states back to zero. The experimental results are depicted in Figures 5 and 6. The stabilization achieved by both proposed fuzzy controllers converge to the desired state faster than the feedback linearization technique. It is worth pointing out that the controller proposed in (Altug et al., 2002) did not consider any type of noise or external disturbances.

The third experiment benchmarks the proposed fuzzy controllers with simulation results of a backstepping controller tuned for optimized nonlinear control of a tethered quadrotor (Bouabdallah et al., 2004b). The backstepping control offers high robustness against large disturbances as compared to the feedback linearization technique which is clear from the results. To test the attitude stabilization of the proposed fuzzy controllers, the system is initialized with severe rotational angles $\Omega^T_{\mathcal{F}_i} = [45, 45, 45]$ degrees. Again, the sensory noise and winds are introduced as disturbances. The experimental results show

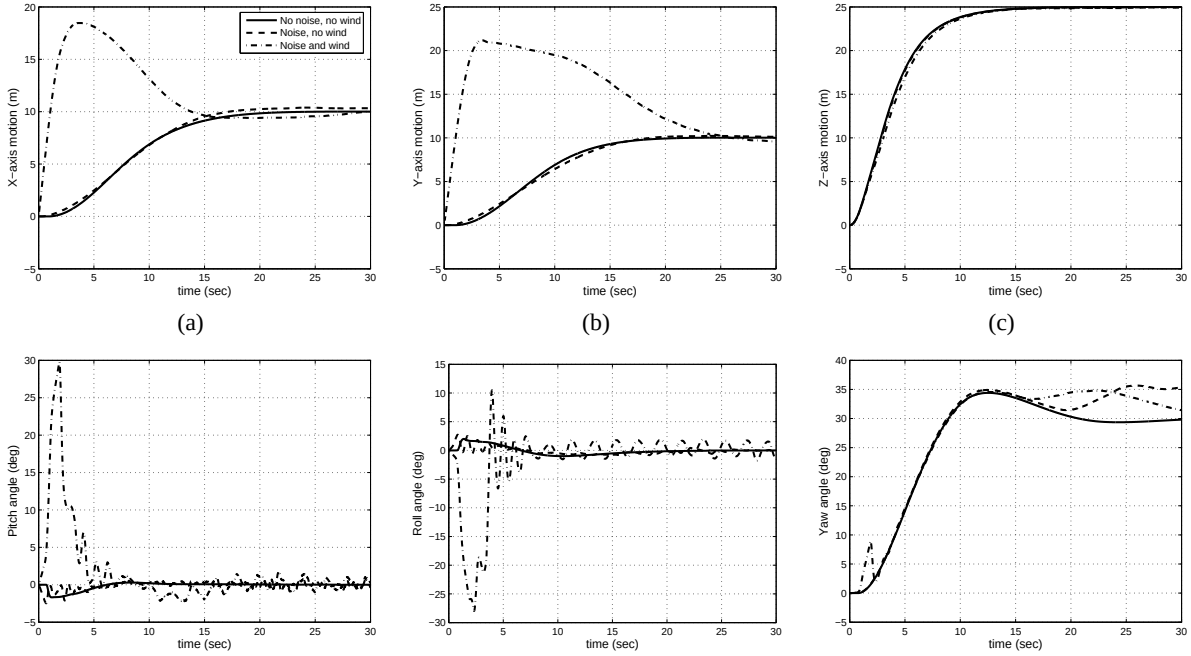


Figure 4: Experiment 1, Sugeno FLC's performance with and without disturbances. Quadrotor states: (a) x-axis; (b) y-axis;; (c) z-axis (altitude); (d) pitch (θ); (e) roll (ϕ); and (f) yaw (ψ).

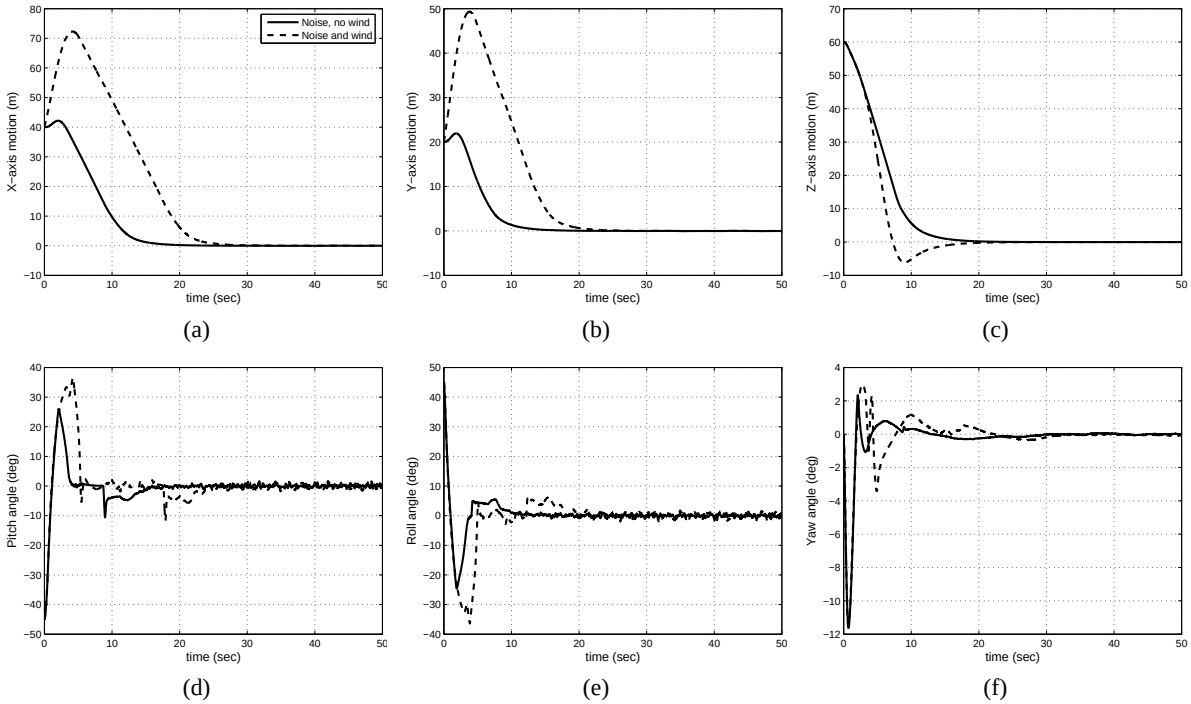


Figure 5: Experiment 2, Mamdani FLC's performance with a different initial condition. Quadrotor states: (a) x-axis; (b) y-axis; (c) z-axis (altitude); (d) pitch (θ); (e) roll (ϕ); and (f) yaw (ψ).

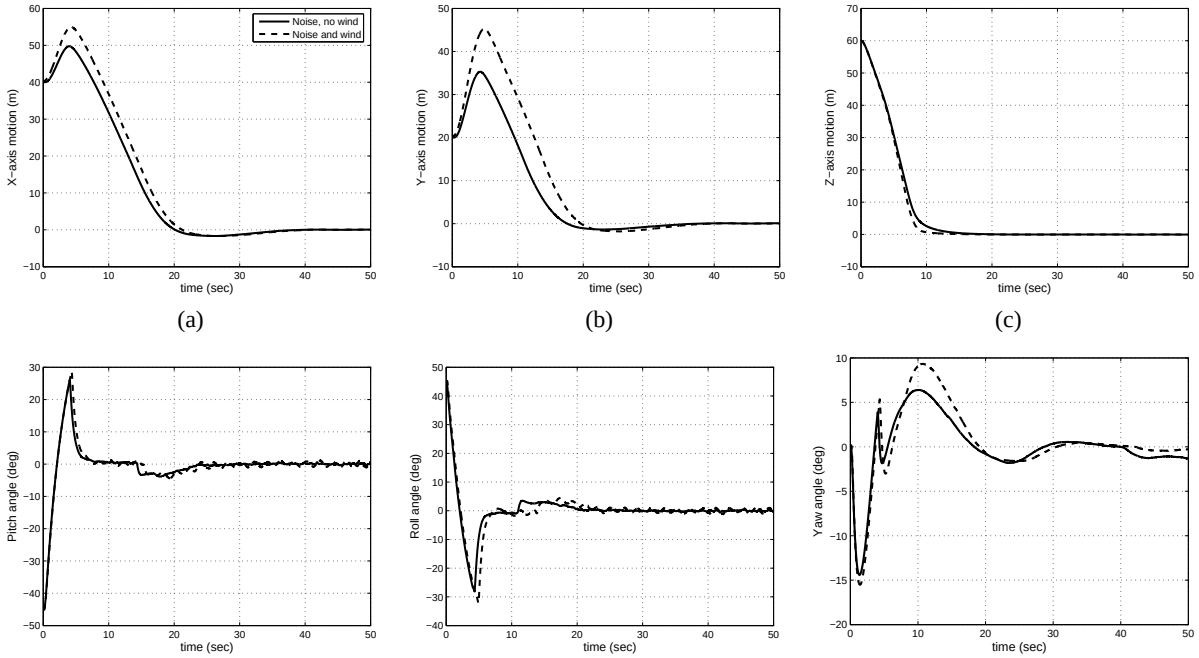


Figure 6: Experiment 2, Sugeno FLC's performance with a different initial condition. Quadrotor states: (a) x-axis; (b) y-axis;; (c) z-axis (altitude); (d) pitch (θ); (e) roll (ϕ); and (f) yaw (ψ).

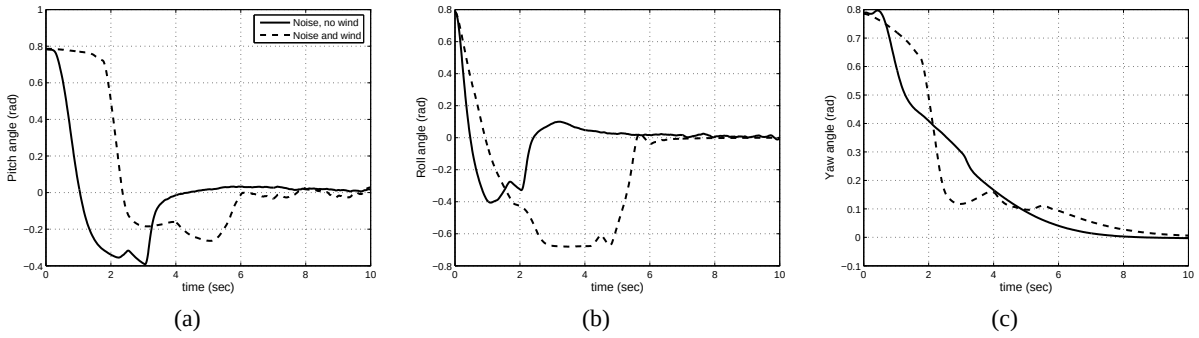


Figure 7: Experiment 3, Mamdani FLC's attitude stabilization. Quadrotor orientation: (a) pitch angle (θ); (b) roll angle (ϕ); and (c) yaw angle (ψ).

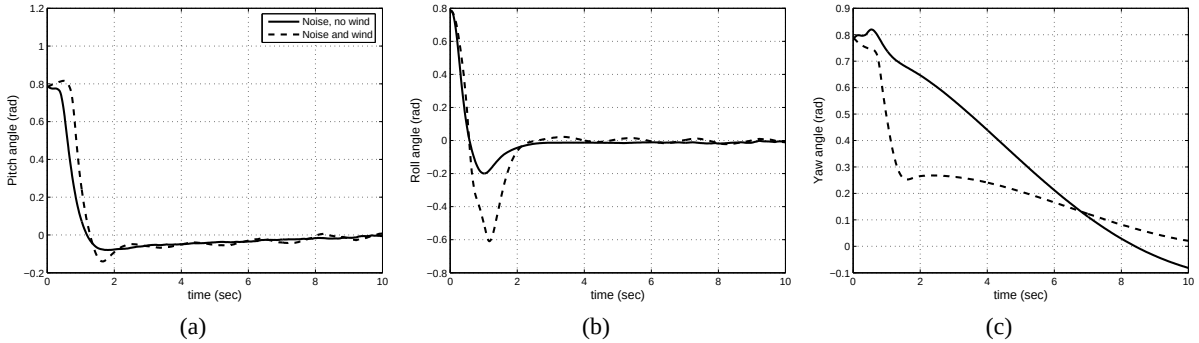


Figure 8: Experiment 3, Sugeno FLC's attitude stabilization. Quadrotor orientation: (a) pitch angle (θ); (b) roll angle (ϕ); and (c) yaw angle (ψ).

the performance of the fuzzy controllers as compared to a robust backstepping technique for nonlinear control in Figures 7, 8. Both Mamdani and Sugeno fuzzy controllers are able to stabilize the quadrotor from the critical initial conditions. The settling times are slightly longer than the backstepping controller, specially for the ψ angle which can be improved with further tuning of the FLC's membership functions. However, this is overlooked because the fuzzy controllers are stabilizing both position and attitude despite the presence of strong wind conditions as high as $30m/s$ and sensory noise of $-55dB$.

5 CONCLUSIONS

A fuzzy logic approach was proposed for the autonomous control of quadrotors without the need for a mathematical model of their complex and ill-defined dynamics. The fuzzy technique was implemented through Sugeno and Mamdani inference engines for comparison. The controller comprises of six individual fuzzy controllers designated for the control of the quadrotor's position and orientation. The investigation on the two types of control methodologies is conducted in a simulation environment, where disturbances such as wind conditions and sensor noise are incorporated for a more realistic simulation. The results demonstrated a successful control performance of the quadrotor with both Mamdani and Sugeno fuzzy controllers despite the disturbances. When compared with other control techniques presented in the literature for the same purpose, the proposed method showed a higher robustness and faster convergence, where satisfactory attitude stabilization is achieved despite stringent initial conditions and severe disturbances. The future work will be directed towards a real-world implementation of the proposed fuzzy controllers. Furthermore, the online adaptation of controllers will be investigated.

REFERENCES

Altug, E., Ostrowski, J., and Mahony, R. (2002). Control of a quadrotor helicopter using visual feedback. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, pages 72–77.

Bouabdallah, S., Murrieri, P., and Siegwart, R. (2004a). Design and control of an indoor micro quadrotor. In *Proceedings of the International Conference on Robotics and Automation*.

Bouabdallah, S., Noth, A., and Siegwart, R. (2004b). PID vs LQ control techniques applied to an indoor micro quadrotor. In *International Conference on Intelligent Robots and Systems*.

Coza, C. and Macnab, C. (2006). A new robust adaptive-fuzzy control method applied to quadrotor helicopter stabilization. In *Annual meeting of the North American Fuzzy Information Processing Society*, pages 454–458.

Dzul, A., Castillo, P., and Lozano, R. (2004). Real-time stabilization and tracking of a four-rotor mini rotorcraft. *IEEE Transaction on Control System Technology*, 12(4):510–516.

Fincannon, T., Barnes, L. E., Murphy, R. R., and Riddle, D. L. (2004). Evidence of the need for social intelligence in rescue robots. In *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1089–1095.

Guenard, N., Hamel, T., and Mahony, R. (2008). A practical visual servo control for an unmanned aerial vehicle. *IEEE Transactions on Robotics*, 24(2):331–340.

Guenard, N., Hamel, T., and Moreau, V. (2005). Dynamic modeling and intuitive control strategy for an X4-flyer. In *International Conference on Control and Automation*, pages 141–146.

Statistics, C. W. (2009). courtesy of environment canada. [Online] Available. http://www.weatherstats.ca/.

Tarbouchi, M., Dunfied, J., and Labonte, G. (2004). Neural network based control of a four rotor helicopter. In *International Conference on Industrial Technology*, pages 1543–1548.

Tayebi, A. and McGilvray, S. (2006). Attitude stabilization of a vtol quadrotor aircraft. *IEEE Transaction on Control System Technology*, 14(3):562–571.

Waslander, S. L., Hoffmann, G. M., Jang, J. S., and Tomlin, C. J. (2005). Multi-agent quadrotor testbed control design: integral sliding mode vs. reinforcement learning. In *International Conference on Intelligent Robots and Systems*, pages 468–473.

SHORT PAPERS

RICCATI SOLUTION FOR DISCRETE STOCHASTIC SYSTEMS WITH STATE AND CONTROL DEPENDENT NOISE

Randa Herzallah
Faculty of Engineering Technology, Al-Balqa' Applied University, Jordan
randa_herzal@hotmail.com

Keywords: Functional uncertainties, Cautious controller, Stochastic systems, Nonlinear functions.

Abstract: In this paper we present the Riccati solution of linear quadratic control problems with input and state dependent noise which is encountered during our previous study to the adaptive critic solution for systems characterized by functional uncertainty. Uncertainty of the system equations is quantified using a state and control dependent noise model. The derived optimal control law is shown to be of cautious type controllers. The derivation of the Riccati solution is via the principle of optimality. The Riccati solution is implemented to linear multi dimensional control problem and compared to the certainty equivalent Riccati solution.

1 INTRODUCTION

The objective of this paper is to introduce the Riccati solution of stochastic linear quadratic systems with input and state dependent noise which is encountered during our previous study (Herzallah, 2007) of the adaptive critic solution to stochastic systems characterized by functional uncertainty. The problem of stochastic linear quadratic control is discussed in (Rami et al., 2001) and is shown to have different form than that of traditional linear quadratic control. However, the work in (Rami et al., 2001) discusses models with multiplicative white noise on both the state and control and it is for continuous time systems. In the current paper the optimal control law for stochastic discrete linear quadratic systems characterized by functional uncertainty will be derived. This yields a cautious type controller which takes into consideration model uncertainty when calculating the optimal control law.

The optimization problem of the linear stochastic control with state and control dependent noise is to find a feedback control which minimizes the following quadratic cost function (Herzallah, 2007):

$$L = \sum_{k=0}^{N-1} U(\mathbf{x}(k),\mathbf{u}(k)) + \psi[\mathbf{x}(N)], \qquad (1)$$

where

$$U(\mathbf{x}(k),\mathbf{u}(k)) = \mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) + \mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k) \qquad (2)$$

$$\psi[\mathbf{x}(N)] = \mathbf{x}^T(N)\mathbf{C}\mathbf{x}(N) + \mathbf{Z}\mathbf{x}(N) + U_o, \qquad (3)$$

subject to the system equation given by

$$\mathbf{x}(k+1) = \tilde{\mathbf{G}}\mathbf{x}(k) + \tilde{\mathbf{H}}\mathbf{u}(k) + \tilde{\eta}(k+1), \qquad (4)$$

where N is the time horizon, $\mathbf{x} \in R^n$ represents the state vector of the system, $\mathbf{u} \in R^m$ denotes the control action, $U(\mathbf{x}(k),\mathbf{u}(k))$ is a utility function, $\psi[(\mathbf{x}(N)$ is the weight of the performance measure due to the final state, and $\tilde{\eta}(k+1)$ is an additive noise signal assumed to have zero mean Gaussian distribution of covariance matrix $\tilde{\mathbf{P}}$. If the matrices $\tilde{\mathbf{G}}$ and $\tilde{\mathbf{H}}$ were known and the system was noiseless, the solution of this problem is well known (Anderson and Moore, 1971; Ogata, 1987). The optimal control is a linear function of $\mathbf{x}$ which is independent of the additive noise $\tilde{\eta}(k+1)$. This solution is also applicable in the presence of independent noise $\tilde{\eta}(k+1)$, because the covariance matrix $\tilde{\mathbf{P}}$, of the noise term is independent of **u**(k).

In the current paper the optimal control for systems with unknown models will be derived. It has been shown that systems with unknown functions should be formulated in an adaptive control framework which is known to have functional uncertainties (Fel'dbaum, 1960; Fel'dbaum, 1961). This means that state and control dependent noise always accompany systems with unknown models. In the literature, three different methods were used to handle the control problem of systems characterized by functional uncertainty. The first method is the heuristic equivalent control method (Åström and Wittenmark, 1989; Guo and Chen, 1991; Xie and Guo, 1998; Yaz, 1986) in which the control is found by solving for the equivalent deterministic system and then simply replace the

unknown variables by their estimates. The second method is the cautious control method (Goodwin and Sin, 1984; Apley, 2004; Campi and Prandini, 2003; Herzallah and Lowe, 2007) which takes the uncertainty of the estimates into consideration when calculating the control but do not plan for any probing signals to reduce the future estimation of uncertainty. The last but most efficient method is the dual control method (Fel'dbaum, 1960; Fel'dbaum, 1961; Fabri and Kadirkamanathan, 1998; Filatov and Unbehauen, 2000; Maitelli and Yoneyama, 1994) which takes uncertainty of the estimates into consideration when estimating the control and at the same time plan to reduce future estimation of uncertainty.

The Riccati solution in this paper is for the more general systems of equation (4), where the parameters of the system equation are unknown and where the noise term is state and control dependent. The parameters of the model are to be estimated on–line based on some observations. Not only the model parameters are to be estimated on–line, but also the state dependent noise which characterizes uncertainty of the parameters estimate and allows estimating the conditional distribution of the system output or state. The conditional distribution of the system output will be estimated by the method used in (Herzallah, 2007). The optimal control is again linear in $\mathbf{x}$, but is now rather critically dependent on the parameters of the estimated uncertainty of the error $\tilde{\eta}(k+1)$. This in turn, yields a cautious type controller which takes into consideration model uncertainty when calculating the optimal control law. A numerical example is provided and the result is compared to the certainty equivalent controller.

The Riccati solution will be introduced soon, but first we give a brief discussion about estimating model uncertainty which we need for the derivation of the Riccati solution of the cautious controller.

2 BASIC ELEMENTS

As a first step to the optimization problem, the conditional distribution of the system output or state needs to be estimated. According to theorem 4.2.1 in (Gersho and Gray, 1992), the minimum mean square error (MMSE) estimate of a random vector $\mathcal{Z}$ given another random vector $\mathcal{X}$ is simply the conditional expectation of $\mathcal{Z}$ given $\mathcal{X}$, $\hat{\mathcal{Z}} = E(\mathcal{Z} \mid \mathcal{X})$. For the linear systems discussed in this paper, a generalized linear model is used to model the expected value of the system output,

$$\hat{\mathbf{x}}(k+1) = \mathbf{G}\mathbf{x}(k) + \mathbf{H}\mathbf{u}(k) \tag{5}$$

The parameters of the generalized linear model are then adjusted using an appropriate gradient based method to optimize a performance function based on the error between the plant and the linear model output. The stochastic model of the system of equation (4) is then shown (Herzallah, 2007) to have the following form:

$$\mathbf{x}(k+1) = \hat{\mathbf{x}}(k+1) + \eta(k+1), \tag{6}$$

where $\eta(k+1)$ represents an input dependent random noise.

Another generalized linear model which has the same structure and same inputs as that of the model output is then used to predict the covariance matrix, $\mathbf{P}$ of the error function $\eta(k+1)$,

$$\mathbf{P} = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k). \tag{7}$$

where $\mathbf{A}$ and $\mathbf{B}$ are partitioned matrices and are updated such that the error between the actual covariance matrix and the estimated one is minimized.

Detailed discussion about estimating the conditional distribution of the system output can be found in (Herzallah, 2007; Herzallah and Lowe, 2007).

3 RICCATI SOLUTION AND MAIN RESULT

In this section we derive the Riccati solution of the infinite horizon linear quadratic control problem characterized by functional uncertainty. We show here that the optimal control law is a state feedback law which depends on the parameters of the estimated uncertainty, and that the optimal performance index is quadratic in the state $\mathbf{x}(k)$ which also dependent on the estimated uncertainty. The derivation is based on the principal of optimality and is for finite horizon control problem which is known to be the steady state solution for an infinite horizon control problem. Hence, by the principal of optimality the objective is to find the optimal control sequence which minimizes Bellman's equation (Bellman, 1961; Bellman, 1962)

$$J[(\mathbf{x}(k)] = U(\mathbf{x}(k), \mathbf{u}(k)) + \gamma < J[\mathbf{x}(k+1)] >, \tag{8}$$

where $< . >$ is the expected value, $J[\mathbf{x}(k)]$ is the cost to go from time k to the final time, $U(\mathbf{x}(k), \mathbf{u}[x(k)])$ is the utility which is the cost from going from time k to time $k+1$, and $< J[\mathbf{x}(k+1)] >$ is assumed to be the average minimum cost from going from time $k+1$ to the final time. The term γ is a discount factor $(0 \leq \gamma \leq 1)$ which allows the designer to weight the relative importance of present versus future utilities.

Using the general expressions of Equations (2), (6) and (5) in Bellman's equation and

taking $\gamma = 1$, yields

$$J[(\mathbf{x}(k)] = \mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k) + \mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) + < J[\mathbf{G}\mathbf{x}(k) + \mathbf{H}\mathbf{u}(k) + \eta(k+1)] > . \quad (9)$$

The true value of the cost function J is shown in (Herzallah, 2007) to be quadratic with the following form

$$J(\mathbf{x}) = \mathbf{x}^T\mathbf{M}\mathbf{x} + \mathbf{S}\mathbf{x} + U_0. \quad (10)$$

Making use of this result in equation (9) yields

$$\begin{aligned} J[(\mathbf{x}(k)] = {} & \mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k) + \mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) \\ & + < [\mathbf{G}\mathbf{x}(k) + \mathbf{H}\mathbf{u}(k) + \eta(k+1)]^T \\ & \mathbf{M}(k+1)[\mathbf{G}\mathbf{x}(k) + \mathbf{H}\mathbf{u}(k) + \eta(k+1)] \\ & + \mathbf{S}(k+1)[\mathbf{G}\mathbf{x}(k) + \mathbf{H}\mathbf{u}(k) + \eta(k+1)] > . \end{aligned} \quad (11)$$

Evaluating the expected value of the third term of equation (11) and using equation (7) yields

$$\begin{aligned} J[(\mathbf{x}(k)] &= \mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k) + \mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) \\ &+ \mathbf{x}^T(k)\mathbf{G}^T\mathbf{M}(k+1)\mathbf{G}\mathbf{x}(k) \\ &+ 2\mathbf{x}^T\mathbf{G}^T\mathbf{M}(k+1)\mathbf{H}\mathbf{u}(k) \\ &+ \mathbf{u}^T(k)\mathbf{H}^T\mathbf{M}(k+1)\mathbf{H}\mathbf{u}(k) \\ &+ \mathrm{tr}[\mathbf{A}\mathbf{M}(k+1)]\mathbf{x}(k) + \mathrm{tr}[\mathbf{B}\mathbf{M}(k+1)]\mathbf{u}(k) \\ &+ \mathbf{S}(k+1)[\mathbf{G}\mathbf{x}(k) + \mathbf{H}\mathbf{u}(k)]. \end{aligned} \quad (12)$$

Minimization of the explicit performance index defined in Equation (12) leads to the control law specified in the following theorem.

Theorem 1: The control law minimizing the performance index $J[(\mathbf{x}(k)]$ of Equation (12), is given by

$$\mathbf{u}^* = -\mathbf{K}(k)\mathbf{x}(k) - \mathbf{Z}(k), \quad (13)$$

where

$$\mathbf{K}(k) = \left[\mathbf{R} + \mathbf{H}^T\mathbf{M}(k+1)\mathbf{H}\right]^{-1}\left[\mathbf{H}^T\mathbf{M}(k+1)\mathbf{G}\right] \quad (14)$$

$$\mathbf{Z}(k) = \left[\mathbf{R} + \mathbf{H}^T\mathbf{M}(k+1)\mathbf{H}\right]^{-1}\frac{1}{2}\left[\mathrm{tr}[\mathbf{B}\mathbf{M}(k+1)] + \mathbf{S}(k+1)\mathbf{H}\right]. \quad (15)$$

Proof. This theorem can be proved directly by deriving Equation (12) with respect to the control and setting the derivative equal to zero. Note that the optimal control law consists of two terms: the linear term in $\mathbf{x}$ which is equivalent to the linear term obtained in deterministic control problems and an extra constant term which gives cautiousness to the optimal control law. Note also that the evaluation of the optimal control law requires evaluating the matrix, $\mathbf{M}(k+1)$ and the vector, $\mathbf{S}(k+1)$. This evaluation can be obtained by evaluating the optimal cost function $J[(\mathbf{x}(k)]$.

The optimal cost function $J[(\mathbf{x}(k)]$ can be obtained by substituting Equation (13) in (12). This yields the quadratic cost function defined in the following theorem.

Theorem 2: The optimal control law defined in Equation (13) yields a quadratic cost function of the following form

$$J[(\mathbf{x}(k)] = \mathbf{x}^T(k)\mathbf{M}(k)\mathbf{x}(k) + \mathbf{S}(k)\mathbf{x}(k) + U_0, \quad (16)$$

where

$$\begin{aligned} \mathbf{M}(k) &= \mathbf{Q} + \mathbf{G}^T\mathbf{M}(k+1)\mathbf{G} \\ &- \mathbf{G}^T\mathbf{M}(k+1)\mathbf{H}\mathbf{F}^{-1}\mathbf{H}^T\mathbf{M}(k+1)\mathbf{G} \end{aligned} \quad (17)$$

$$\begin{aligned} \mathbf{S}(k) &= \mathrm{tr}[\mathbf{A}\mathbf{M}(k+1)] - \Big\{\mathrm{tr}[\mathbf{B}\mathbf{M}(k+1)] \\ &+ \mathbf{S}(k+1)\mathbf{H}\Big\}\mathbf{F}^{-1}\mathbf{H}^T\mathbf{M}(k+1)\mathbf{G} \\ &+ \mathbf{S}(k+1)\mathbf{G}. \end{aligned} \quad (18)$$

and where

$$\mathbf{F}^{-1} = [\mathbf{R} + \mathbf{H}^T\mathbf{M}(k+1)\mathbf{H}]^{-1}. \quad (19)$$

Equation (17) is called the Riccati equation. It is similar to that obtained for the certainty equivalent controller. Equation (18) is dependent on the solution of the Riccati equation. It provides cautiousness to the optimal quadratic controller, therefore, will be referred to as the equation of cautiousness. According to equation (3), the optimal cost at $k = N$ equal to $\psi[(\mathbf{x}(N)]$. This means that $\mathbf{M}(N) = \mathbf{C}$ and $\mathbf{S}(N) = \mathbf{Z}$. Hence equation (17) and (18) can be solved uniquely backward from $k = N$ to $k = 0$. That is $\mathbf{M}(N), \mathbf{M}(N-1), \ldots, \mathbf{M}(0)$ and $\mathbf{S}(N), \mathbf{S}(N-1), \ldots, \mathbf{S}(0)$ can be obtained starting from $\mathbf{M}(N)$ and $\mathbf{S}(N)$ which are known.

To reemphasize, the matrix $\mathbf{M}(k)$ has an equivalent form similar to that obtained for deterministic control problems. However the optimal control law and the cost function are dependent on the values of the vector $\mathbf{S}(k)$ as well as on $\mathbf{M}(k)$.

For infinite horizon control problems, the optimal control solution becomes a steady state solution of the finite horizon control (Ogata, 1987; Anderson and Moore, 1971). Hence $\mathbf{K}(k)$, $\mathbf{Z}(k)$, $\mathbf{M}(k)$, and $\mathbf{S}(k)$ become constant and defined as follows

$$\mathbf{K} = \left[\mathbf{R} + \mathbf{H}^T\mathbf{M}\mathbf{H}\right]^{-1}\left[\mathbf{H}^T\mathbf{M}\mathbf{G}\right] \quad (20)$$

$$\mathbf{Z} = \left[\mathbf{R} + \mathbf{H}^T\mathbf{M}\mathbf{H}\right]^{-1}\frac{1}{2}\left[[\mathrm{tr}\mathbf{B}\mathbf{M}] + \mathbf{H}\mathbf{S}\right] \quad (21)$$

$$\mathbf{M} = \mathbf{Q} + \mathbf{G}^T\mathbf{M}\mathbf{G} - \mathbf{G}^T\mathbf{M}\mathbf{H}\mathbf{F}^{-1}\mathbf{H}^T\mathbf{M}\mathbf{G} \quad (22)$$

$$\begin{aligned} \mathbf{S} &= \mathrm{tr}[\mathbf{A}\mathbf{M}] - \left[\mathrm{tr}[\mathbf{B}\mathbf{M}] + \mathbf{S}\mathbf{H}\right]\mathbf{F}^{-1}\mathbf{H}^T\mathbf{M}\mathbf{G} \\ &+ \mathbf{S}\mathbf{G}. \end{aligned} \quad (23)$$

In implementing the steady state optimal controller, the steady state solution of the Riccati equation as

well as the equation of cautiousness should be obtained. Since the Riccati equation of the cautious controller derived in this paper has similar form to that of the certainty equivalent controller, standard methods proposed in (Ogata, 1987) can be implemented to obtain the solution of the Riccati equation. To obtain the solution of the steady state equation of cautiousness given by Equation (23),

$$\mathbf{S} = \mathrm{tr}[\mathbf{AM}] - \Big\{\mathrm{tr}[\mathbf{BM}] + \mathbf{SH}\Big\}\mathbf{F}^{-1}\mathbf{H}^T\mathbf{MG} + \mathbf{SG},$$

we simply start with the non steady state equation of cautiousness which was given by Equation (18),

$$\begin{aligned}\mathbf{S}(k) &= \mathrm{tr}[\mathbf{AM}(k+1)] - \Big\{\mathrm{tr}[\mathbf{BM}(k+1)] \\ &+ \mathbf{S}(k+1)\mathbf{H}\Big\}\mathbf{F}^{-1}\mathbf{H}^T\mathbf{M}(k+1)\mathbf{G} \\ &+ \mathbf{S}(k+1)\mathbf{G}, \end{aligned} \quad (24)$$

by substituting the steady state matrix $\mathbf{M}$ and reversing the direction of time, we modify Equation (24) to read

$$\begin{aligned}\mathbf{S}(k+1) &= \mathrm{tr}[\mathbf{AM}] - \Big\{\mathrm{tr}[\mathbf{BM}] \\ &+ \mathbf{S}(k)\mathbf{H}\Big\}\mathbf{F}^{-1}\mathbf{H}^T\mathbf{MG} + \mathbf{S}(k)\mathbf{G}. \end{aligned} \quad (25)$$

Then beginning the solution with $\mathbf{S}(0) = \mathbf{0}$, iterate Equation (25) until a stationary solution is obtained.

4 SIMULATION EXAMPLE

To numerically test and demonstrate the Riccati solution of the cautious controller, the theory developed in the previous section is applied here to 2–inputs 3–outputs control problem described by the following stochastic equation

$$\mathbf{x}(k+1) = \mathbf{Gx}(k) + \mathbf{Hu}(k) + \mathbf{w}(k+1), \quad (26)$$

where

$$\mathbf{G} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -0.16 & 0.84 & 0 \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix},$$

$$\begin{bmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \end{bmatrix} = \begin{bmatrix} 5 \\ -3 \\ 2 \end{bmatrix}$$

$$E[\mathbf{w}(k+1)\mathbf{w}^T(k+1)] = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}.$$

Note that although the added noise to the system of Equation (26) is not state and control dependent, the estimated noise is state and control dependent reflecting the fact that the estimated model is not exact. The performance index to be minimized is specified so as to keep the state values near the origin. That is

$$J = \sum_{k=0}^{\infty} [\mathbf{x}^T(k)\mathbf{Qx}(k) + \mathbf{u}^T(k)\mathbf{Ru}(k)], \quad (27)$$

where $\mathbf{Q} = \mathbf{I}$ and $\mathbf{R} = \mathbf{I}$. Three generalized linear models were used to provide a prediction for the states, x_1, x_2 and x_3. The covariance matrix of the state vector is assumed to be diagonal, therefore, one generalized linear model with three outputs was used to provide a prediction for the variance of the error of estimating the first state x_1, the second state x_2 and the third state x_3.

For comparison purposes, the optimal control law is calculated by assuming the certainty equivalence method where conventional Riccati solution is used to estimate the optimal control, and by taking uncertainty measure into consideration where the proposed Riccati solution is used to estimate the optimal control. The same noise sequence and initial conditions were used in each case. The generalized linear models were never subjected to an initial off–line training phase. Closed loop control was activated immediately, with the initial parameter estimates selected at random from a zero mean, isotropic Gaussian, with variance scaled by the fan-in of the output units. The output of the two methods is shown in Figure 1. As expected, the figure shows that the certainty equivalence controller initially responds crudely, exhibiting large transient overshoot because it is not taking into consideration the inaccuracy of the parameter estimates. Only after the initial period, when the parameters of the system model converge, does the control assume good tracking. On the other hand, the cautious controller does not react hastily during the initial period, knowing that the parameters estimate are still inaccurate.

5 CONCLUSIONS

In this paper, we have derived the Riccati solution for more general systems with unknown functionals and state dependent noise. The Riccati solution of this paper is suitable for deterministic and stochastic control problems characterized by functional uncertainty. The optimal control is of cautious type and takes into consideration model uncertainty.

The derived Riccati equation in this paper has similar form to that of the certainty equivalent controller. However, as a result of considering the uncertainty on the models, the derived optimal control law has been

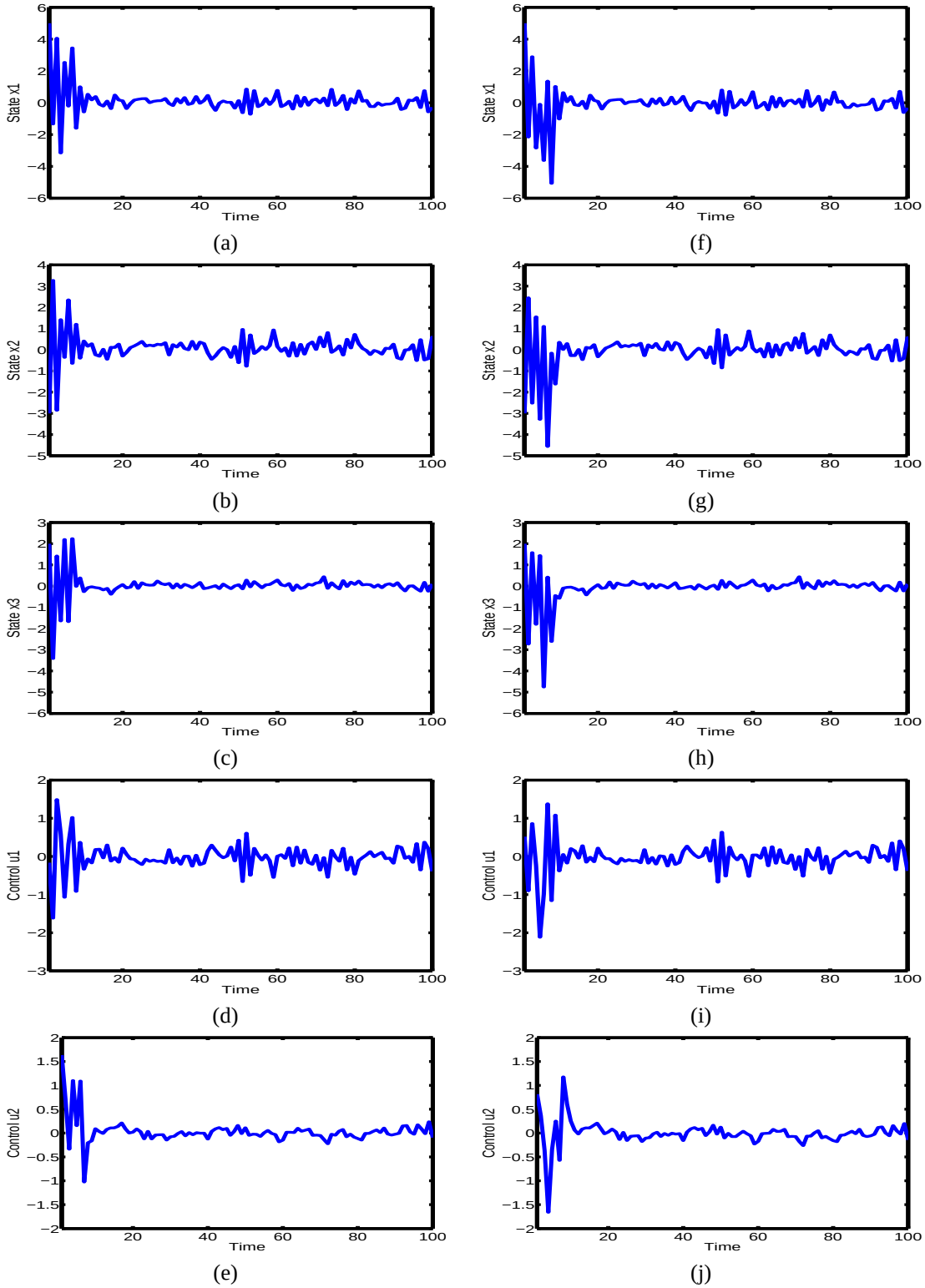


Figure 1: Controlled multi dimensional stochastic system: (a) State 1 using the proposed Riccati solution. (b) State 2 using the proposed Riccati solution. (c) State 3 using the proposed Riccati solution. (d) Control 1 using the proposed Riccati solution. (e) Control 2 using the proposed Riccati solution. (f) State 1 using the conventional Riccati solution. (g) State 2 using the conventional Riccati solution. (h) State 3 using the conventional Riccati solution. (i) Control 1 using the conventional Riccati solution. (j) Control 2 using the conventional Riccati solution.

shown to have an extra term which depends on the estimated uncertainty. A simulation example has illustrated the efficacy of the cautious controller.

REFERENCES

Anderson, B. D. O. and Moore, J. B. (1971). *Linear Optimal Control*. Prentice-Hall, Inc., Englewood Cliffs, N.J.

Apley, D. (2004). A cautious minimum variance controller with ARIMA disturbances. *IIE Transactions*, 36(5):417–432.

Åström, K. J. and Wittenmark, B. (1989). *Adaptive Control*. Addison-Wesley, Reading, MA, U.S.A.

Bellman, R. E. (1961). *Adaptive Control Processes*. Princeton University Press, Princeton, NJ.

Bellman, R. E. (1962). *Applied Dynamic Programming*. Princeton University Press, Princeton, NJ.

Campi, M. C. and Prandini, M. (2003). Randomized algorithms for the synthesis of cautious adaptive controllers. *Systems and Control Letters*, 49:21–36.

Fabri, S. and Kadirkamanathan, V. (1998). Dual adaptive control of nonlinear stochastic systems using neural networks. *Automatica*, 34(2):245–253.

Fel'dbaum, A. A. (1960). Dual control theory I-II. *Automation and Remote Control*, 21:874–880.

Fel'dbaum, A. A. (1961). Dual control theory III-IV. *Automation and Remote Control*, 22:109–121.

Filatov, N. M. and Unbehauen, H. (2000). Survey of adaptive dual control methods. *IEE Proceedings Control Theory and Applications*, 147:118–128.

Gersho, A. and Gray, R. M. (1992). *Vector Quantization and Signal Compression*. Kluwer Academic Publishers.

Goodwin, G. C. and Sin, K. S. (1984). *Adaptive Filtering Prediction and Control*. Prentice-Hall, Englewood Cliffs, NJ.

Guo, L. and Chen, H.-F. (1991). The Åström–Wittenmark self–tuning regulator revisited and ELS–based adaptive trackers. *IEEE Transactions on Automatic Control*, 36(7):802–812.

Herzallah, R. (2007). Adaptive critic methods for stochastic systems with input-dependent noise. *Automatica*, 43(8):1355–1362.

Herzallah, R. and Lowe, D. (2007). Distribution modeling of nonlinear inverse controllers under a Bayesian framework. *IEEE Transactions on Neural Networks*, 18:107–114.

Maitelli, A. L. and Yoneyama, T. (1994). A two stage suboptimal controller for stochastic systems using approximate moments. *Automatica*, 30(12):1949–1954.

Ogata, K. (1987). *Discrete Time Control Systems*. Prentice Hall, Inc., EnglewoodCliffs, New Jersey.

Rami, M. A., Chen, X., Moore, J. B., and Zhou, X. Y. (2001). Solvability and asymptotic behavior of generalized riccati equations arising in indefinite stochastic lq controls. *IEEE Transactions on Automatic Control*, 46(3):428–440.

Xie, L. L. and Guo, L. (1998). Adaptive control of a class of discrete time affine nonlinear systems. *Systems and Control Letters*, 35:201–206.

Yaz, E. (1986). Certainty equivalent control of stochastic sytems: Stability property. *IEEE Transactions on Automatic Control*, 31(2):178–180.

INTELLIGENT FAULT DIAGNOSIS USING SENSOR NETWORK

Haris M. Khalid, Rajamani Doraiswami[1]
Systems Engg. Department,King Fahd University of Petroleum and Minerals, Dhahran 31261, Kingdom of Saudi Arabia
[1]Department of Electrical and Computer Engineering, University of New Brunswick, Fredericton, New Brunswick, Canada
g200702310@kfupm.edu.sa, dorai@unb.ca

Lahouari Cheded
Systems Engg. Department, King Fahd University of Petroleum and Minerals, Dhahran 31261, Kingdom of Saudi Arabia
cheded@kfupm.edu.sa

Keywords: Incipient faults, Holistic approach, Fault diagnosis, Model based, Integrated approach.

Abstract: An intelligent diagnostic scheme using sensor network for incipient faults is proposed using a holistic approach which integrates model-, fuzzy logic-, neural network- based schemes. In case the system is highly non-linear and there are enough training data available, a neural network based scheme is preferred; where the rules relating the input and output can be derived, a Fuzzy-logic approach is chosen; and where a model is available, a linearized model is employed. These three schemes are integrated sequentially ensuring thereby that critical information about the presence or absence of a fault is monitored in the shortest possible time, and the complete status regarding the fault is unfolded in time. The proposed scheme is evaluated extensively on simulated examples and on a physical system exemplified by a benchmarked laboratory-scale two-tank system to detect and isolate faults including sensor, actuator and leakage ones.

1 INTRODUCTION

Fault is an undesirable factor in any process control industry. It affects the efficiency of system operation and reduces economic benefit to the industry. The early detection and diagnosis of faults in mission critical systems becomes highly crucial for preventing failure of equipment, loss of productivity and profits, management of assets, reduction of shutdowns, condition-based monitoring, product quality, process reliability, economy, potential hazards, pollution, and conservation of scarce resources. In a chemical industry, the release of hazardous chemicals into the environment requires quick action to limit the harmful impact of such a release. Of much concern is the purposeful release of chemicals in order to cause harm. Quickly detecting and identifying an unknown threat caused by a fault is pivotal to limiting harm and possibly saving lives. Because of the large area covered in either a process control industry or water distribution systems, a single technique is not able to monitor all of the activity in the area of concern. For this reason, a precise pool of intelligent approaches is being developed to create a better response plan. There must be a way to process and clearly present an accurate picture of the fault threat. Information about the constraints associated with an early detection of hazardous material in the environment help shape the proposed methodology, and is one of the main motivations for embedding intelligent tools in diagnosis and decision making (R.J. Patton, 2000).

The purpose of this paper is to present and advance a new methodology for the intelligent detection of incipient faults. New methods of assimilating information from highly complex and nonlinear physical systems with various nonlinearities are being developed. Intelligent tools that have the ability to adapt, such as neural networks and fuzzy inference systems, are brought to bear on both of these aims. Data from a benchmarked laboratory-scale two-tank system is used and the proposed approach evaluated.

The faults include sensor, actuator and leakage faults, and they can be classified broadly as either parametric faults or additive ones. An additive fault manifests itself as an additive exogenous signal in the measured data, while a parametric fault induces a

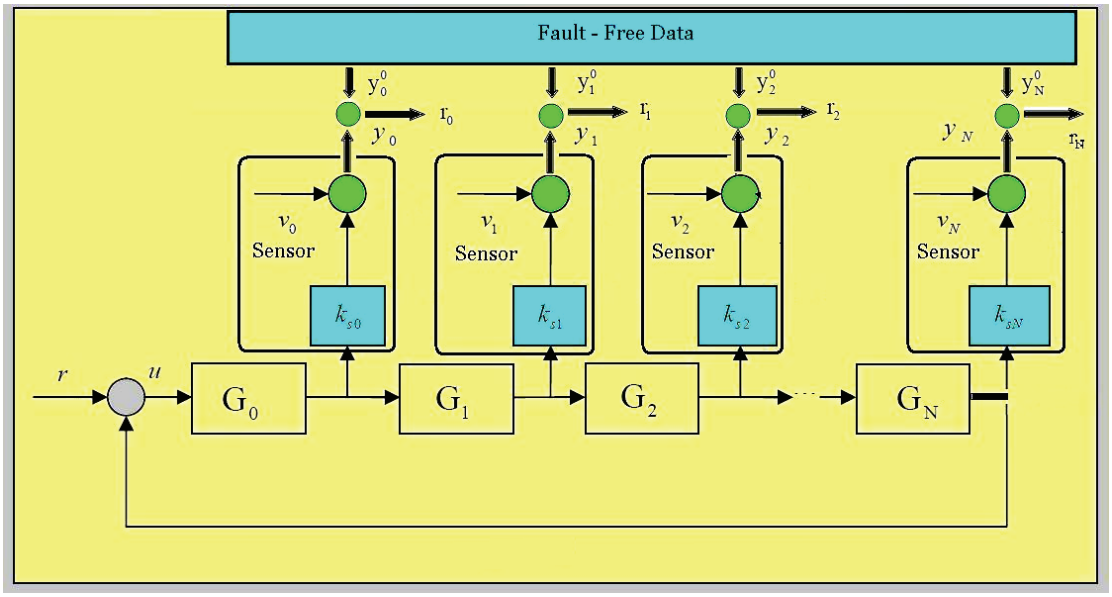


Figure 1: Sensor Network.

variation in the system parameters.

The fault diagnosis scheme can be carried out using a neural network, or fuzzy logic or a model-based technique (L. B. Palma, 2003). While neutral networks can be used to quickly and correctly classify a particular fault, they cannot unravel it and point out its root causes. However, these root causes can be uncovered by supplementing the neural network used by a fuzzy logic scheme, which through the very makeup of its rules, will accurately , albeit more slowly than the neural network, pinpoint the cause(s) that spawned this fault. The synergistic value of this integration will no doubt provide a powerful fault detection scheme. The neural net and fuzzy logic approaches are not geared for the diagnosis of incipient faults, hence the need for, and the inclusion of, a model-based scheme.

2 A SENSOR NETWORK PARADIGM FOR FAULT DIAGNOSIS

A new scheme is proposed here whereby a sensor network paradigm is applied to fault diagnosis. A typical system including a process control system, a water distribution system formed of tanks and network of pipes, a power utility formed of generators and transmission lines, a communication network, and petrochemical industries consisting of a number of control loops, including controllers, sensors and actuators, and various processing plants, as shown in Fig. 1. As such, such a large system will include a sensor network.

A sensor is modelled by a gain and an additive noise, as given below:

$$y_i = k_{si} y_i^0 + v_i \tag{1}$$

where y_{si}, y_{si}^0 and v_i are the measured sensor output, true or fault-free output and additive noise, respectively. Here the gain is such that $0 \le k_{si} \le 1$, with the degree of the fault ranging from no fault at all for $k_{si} = 1$ to a complete failure for $k_{si} = 0$. The subsystems such as actuators, processors and controllers are denoted by transfer functions, G_i. Many systems consisting of several closed loops, each with its own reference input, can be viewed as a sensor network that can be described by a ring-type topology.

The objective of the sensor network is to diagnose faults in both the sensors, through the gains k_{si} and in the subsystems G_i by monitoring the sensor outputs y_i.

The mathematical relations governing the sensor outputs y_i to the input to G_0, denoted by e are:

$$\begin{aligned} y_1 &= G_0 k_{s0} e + v_0 \\ y_2 &= G_0 G_1 k_{s1} e + v_1 \\ y_3 &= G_0 G_1 G_2 k_{s2} e + v_2 \\ &\cdot \\ &\cdot \\ y_i &= G_0 G_1 G_2 \ldots G_{i-1} k_{s(i-1)} e + v_{i-1} \end{aligned} \tag{2}$$

where $e = r - y$.

3 FUZZY LOGIC-BASED FAULT DIAGNOSIS

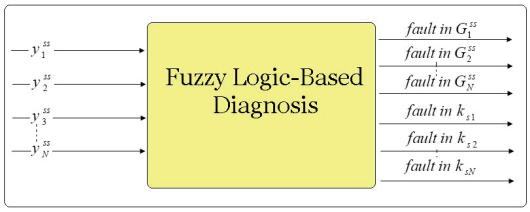


Figure 2: Fuzzy Logic-Based Fault Diagnosis Scheme.

The fuzzy fault diagnosis scheme uses the steady-state values of the sensor outputs, y_i, denoted by y_i^{ss}. A change in the gain k_{si} or a change in the steady-state gain of the transfer function G_i, denoted by G_i^{ss}, is indicative of a fault in the i-th sensor and i-th subsystem, respectively (see Fig.2). Assuming that the noise term is subsumed in the fuzzy membership function, the steady-state model takes the form:

$$y_i^{ss} = G_0^{ss} G_1^{ss} G_2^{ss} \ldots G_{i-1}^{ss} k_{s(i-1)} e \qquad (3)$$

Let us now define linguistic variables such as *zero*, and *non-zero*. For simplicity, we will consider the case where only one device can be faulty at any given time, i.e. the fault is assumed to be simple. In this case, the fuzzy rules may take the following form:

Rule 1: If y_i^{ss} is non-zero, then there is a fault in the steady-state gain G_0^{ss} or G_1^{ss} or G_2^{ss} or…or G_i^{ss} or ith sensor gain k_{si}

Rule II: If y_i^{ss} is zero, then there is no fault in the subsystem's steady-state gain G_0^{ss} or G_1^{ss} or G_2^{ss} or…or G_i^{ss} or ith sensor gain k_{si}

Rule III: If y_i^{ss} is zero and $y_{s(i+1)}$ is non-zero then there is a fault in subsystem G_{i+1}^{ss} or sensor $k_{s(i+1)}$

Rule IV: If y_i^{ss} is non-zero and $y_{s(i+1)}$ is zero then there is a fault in sensor k_{si}

These rules may be generalized to multiple faults.

4 NEURAL NETWORK-BASED FAULT DIAGNOSIS

A fault in the sensor, k_{si}, and or in a subsystem, G_i, can also be diagnosed by using a neural network, as shown in Fig.3. The inputs to the neural network are the spectrum of the coherence between the fault-free and measured sensor outputs.

$$c\left(y_i^0(j\omega), y_i(j\omega)\right) = \frac{\left|y_i^0(j\omega)\, y_i^*(j\omega)\right|^2}{\left|y_i^0(j\omega)\right|^2 \left|y_i(j\omega)\right|^2} \qquad (4)$$

where ω is the frequency in rad/sec, and $c\left(y_i^0(j\omega) y(j\omega)\right)$ is the coherence spectrum. If there is no fault, then $c\left(y_i^0(j\omega) y(j\omega)\right) = 1$ for all frequencies. If the measured and fault-free outputs are incoherent with each other at some frequencies, then the coherence spectrum will be less than 1 at those frequencies.

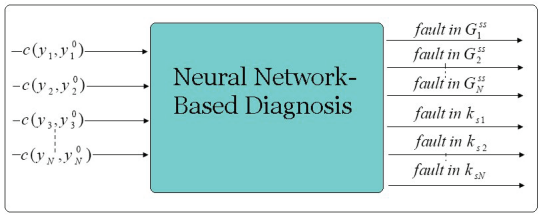


Figure 3: Neural Network-Based Fault Diagnosis Scheme.

5 MODEL-BASED FAULT DIAGNOSIS

A bank of Kalman filters is employed to detect faults. An i-th Kalman filter will be driven by the signal $e(k)$, and the output of the i-th sensor output y_i,

$$\begin{aligned} x_i(k+1) &= A_i x_i(k) + B_i e(k-d) + K_i\left(y_i(k) - \hat{y}_i(k)\right) \\ \hat{y}_i(k) &= C_i \hat{x}_i(k) \end{aligned} \qquad (5)$$

where d is the delay, $\hat{x}_i$ is the estimate of the state, x_i, (A_i, B_i, C_i) is the state-space model of the system with input $e(k)$ and the sensor output, $y_i(k)$. The above-defined Kalman filter is applied to the following transfer function model of the collection of *i* sub-systems:

$$y_i = G_0 G_1 G_2 \ldots G_{i-1} k_{s(i-1)} e + v_{i-1} \qquad (6)$$

5.1 Kalman Filter Design

Let us consider a generic Kalman filter for a system with input u and output, y. The Kalman filter is

designed for the normal fault-free operation. The model of the fault-free system is given by:

$$x(k+1) = A_0 x(k) + B_0 u(k-d) + w(k)$$
$$y(k) = C_0 x(k) + \upsilon(k) \quad (7)$$

Where (A_0, B_0, C_0) are the system matrices obtained from the fault-free system model, $w(k)$ and $\upsilon(k)$ are the zero-mean white plant and measurement noise signals, respectively, with covariances:

$$Q = E\left[w(k)w^T(k)\right], \text{ and } R = E\left[\upsilon(k)\upsilon^T(k)\right] \quad (8)$$

The plant noise, $w(k)$, is a mathematical artifice introduced to account for the uncertainty in the *a-priori* knowledge of the plant model. The larger the covariance Q is, the less accurate the model (A_0, B_0, C_0) is and vice versa.

The Kalman filter is given by:

$$\hat{x}(k+1) = A_0\hat{x}(k) + B_0 u(k-d) + K_0\left(y(k) - C_0\hat{x}(k)\right)$$
$$e(k) = y(k) - C_0\hat{x}(k) \quad (9)$$

where d is the delay and $e(k)$ the residual.

The system model has a pure time delay which is incorporated in the Kalman filter formulation. The Kalman filter estimates the states by fusing the information provided by the measurement $y(k)$ and the *a-priori* information contained in the model, (A_0, B_0, C_0). This fusion is based on the *a priori* information of the plant and the measurement noise covariances, *Q*, and *R*, respectively. When *Q* is small, implying that the model is accurate, the state estimate is obtained by weighting the plant model more than the measurement one. The Kalman gain, K_0, will then be small. On the other hand, when *R* is small implying that the measurement model is accurate, the state estimate is then obtained by weighting the measurement model more than the plant one. The Kalman gain, K_0, will then be large in this case.

The larger K_0 is, the faster the response of the filter will be and the larger the variance of the estimation error becomes. Thus, there is a trade-off between a fast filter response and a small covariance of the residual. An adaptive on-line scheme is employed to tweak the *a- priori* choice of the covariance matrices so that an acceptable trade-off between the Kalman filter performance and the covariance of the residual is reached.

5.2 Fault Isolation

Let e_i be the residual of the i-th Kalman filter. A fault in G_0, G_1 G_2 …or G_i or k_{si} is indicated if the absolute mean of the residual exceeds a specified threshold σ_{th} .

Let us define a 2(N+1) by1 vector of zeros and ones.

$$b_i = \left[g_0 \;\; g_1 \;\; g_2 \;\; \ldots \;\; g_N \;\; \middle| \;\; \kappa_0 \;\; \kappa_1 \;\; \kappa_2 \;\; \ldots \;\; \kappa_N\right] \quad (10)$$

$$g_i = \begin{cases} 0 & no\ \ fault\ in\ G_i \\ 1 & fault\ in\ G_i \end{cases} \quad (11)$$

$$\kappa_i = \begin{cases} 0 & no\ \ fault\ in\ k_{si} \\ 1 & fault\ in\ k_{si} \end{cases} \quad (12)$$

Case I.

If the absolute mean of the i-th residual exceeds the threshold σ_{th}, then b_i will be:

$$b_i = \left[1 \;\; 1 \;\; 1 \;\; X \;\; X....X \;\middle|\; X \;\; X \;\; 1 \;\; X \;\; X...X\right] \quad (13)$$

where X is a don't care value (0 or 1).

If the absolute mean of the (i+1)-st residual does not exceed the threshold σ_{th}, then b_{i+1} will be:

$$b_{i+1} = \left[0 \;\; 0 \;\; 0 \;\; 0 \;\; X....X \;\middle|\; X \;\; X \;\; X \;\; 0 \;\; X...X\right] \quad (14)$$

The intersection between the 2 binary sets b_i and b_{i+1}, amounting to an element-wise binary logical ANDing of these 2 sets, will then clearly indicate that the sensor k_{si} is the faulty one.

Case II.

If the absolute mean of the ith residual does not exceed the specified threshold σ_{th}, then b_i will be:

$$b_i = \left[0 \;\; 0 \;\; 0 \;\; X \;\; X....X \;\middle|\; X \;\; X \;\; 0 \;\; X \;\; X...X\right] \quad (15)$$

If the absolute mean of the (i+1)-st residual exceeds the specified threshold σ_{th}, then b_{i+1} will be:

$$b_{i+1} = \left[1 \;\; 1 \;\; 1 \;\; 1 \;\; X....X \;\middle|\; X \;\; X \;\; X \;\; 1 \;\; X...X\right] \quad (16)$$

This shows that the intersection between the 2 binary sets b_i and b_{i+1}, amounting to an element-wise binary logical ANDing of these 2 sets, will then clearly indicate that either the sensor $k_{s(i+1)}$ or subsystem $G_{s(i+1)}$ is the faulty one.

6 EVALUATION OF THE PROPOSED SCHEME

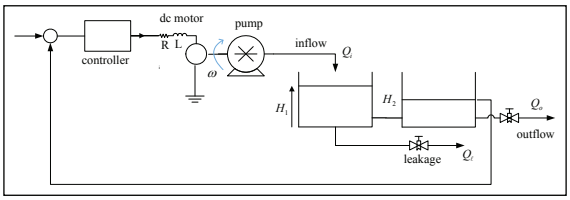


Figure 4: Two-tank Fluid System.

An evaluation of the proposed scheme for fault diagnosis was performed on a benchmark laboratory-scale process control system using National Instruments LABVIEW as shown in Fig 4. Fault diagnosis in a fluid system has enjoyed an increasing importance and popularity in recent years from the points of view of economy, safety, pollution, and conservation of scarce resources (Marco Ferrente, 2008) (Zhang Sheng, 2004)(Doraiswami, 1996) (R.J. Patton, 2000)(Astrom et.,al, 2001) (C. De Persis, 2000)(H. Hammouri, 2002) (K.M. Kinnaert, 1999).

The proposed scheme is used to detect and isolate a fault by a sequential integration of model-free and model-based approaches.

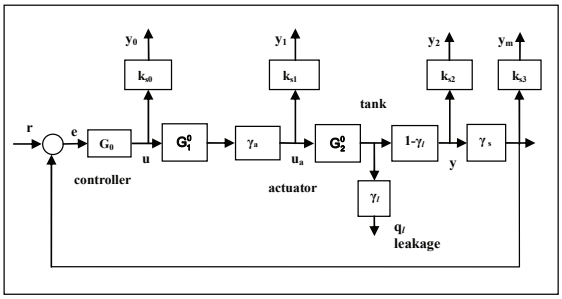


Figure 5: Fluid system subject to a leakage.

We will use a set of fuzzy logic rules to detect a leakage. The fuzzy IF and THEN rules for the two-tank fluid system are derived using the sensor network shown in Fig.1. For the fault diagnosis problem, the equivalent of Fig. 1, is shown in Fig. 5 whose various sub-systems and sensor blocks are all explained below. First, note that the first two blocks in Fig. 5, i.e. G_0 and $G_1 = G_1^0 \gamma_a$, represent the controller and the actuator sub-systems, respectively. As shown in Fig. 5, the leakage is modelled by the gain γ_ℓ which is used to quantify the amount of flow lost from the tank. Thus the net outflow is quantified by the gain $(1-\gamma_\ell)$. Since the two blocks G_2^0 and $(1-\gamma_\ell)$ cannot be dissociated from each other, they are fused into a single block labelled $G_2 = G_2^0(1-\gamma_\ell)$. The feedback sensor, modelled by the gain k_{sf}, is used to feed the plant output y back to the controller, and is modelled by the last block G_3 in Fig. 3, where $G_3 = k_{sf}$. An additional sensor, termed as the redundant sensor of gain k_{s2}, is used here to discriminate between faults in the height sensor and feedback sensor. Even though the control input u does not necessitate a separate sensor to monitor its output as it is freely available from the digital controller (G_0), a separate unit gain, labelled $k_{s0} = 1$, is attributed to it. Similarly, the last sensor, used to monitor the feedback sensor output, is also attributed a unit gain, i.e. $k_{s3} = 1$. The reason for adding these two unit gains to Fig. 5 is motivated by our desire to make the overall sensor network structure for the leakage detection problem fit in well within the general sensor network-based fault detection paradigm shown in Fig. 3. By doing so, the two fuzzy rules (Rules 1 and 2 given earlier) can be readily applied to Fig. 5. The four residuals, r_0, r_1, r_2 and r_3, are the deviations between the fault-free and fault-bearing measurements of the control input , flow rate, height from the redundant sensor, and height from the feedback sensor, respectively.

6.1 Fault Diagnosis using a Model-free Approach

A sequential integration of an artificial neural network (ANN) and a fuzzy logic (FL) approach is employed here to isolate faults.

Fuzzy-logic Approach. The features were chosen to be the steady-state values of the control input, u_{ss}, measured flow flw_{ss} and height h_{ss} values and their fault-free counterparts, u_{ss}^0, flw_{ss}^0 and h_{ss}^0, respectively. The fuzzy logic rules pertinent to this case are similar to those described earlier.

The steady-state gain relating flw_{ss} and u_{ss} is given by:

$$flw_{ss} = G_0^{ss} G_1^{ss} u^{ss} \tag{17}$$

The steady-state gain relating h_{ss} and u_{ss} is given by:

$$h_{ss} = G_0^{ss} G_1^{ss} G_2^{ss} u^{ss} \tag{18}$$

Where $G_0^{ss}, G_1^{ss}, G_2^{ss}$ are the steady-state gains of the actuator, the transfer function relating the control input to the flow, and the transfer function relating the flow to the height, respectively.

The fuzzy IF-and-THEN rules given in the previous section can isolate a leakage from faults in the actuator, flow and height (or level) sensor.

Neural Network Approach. A neural network is driven by the coherence spectrum between the measured height h and the corresponding fault-free one h^0. This coherence spectrum is defined by:

$$c\left(h^0(j\omega), h(j\omega)\right) = \frac{\left|h^0(j\omega)h^*(j\omega)\right|^2}{\left|h^0(j\omega)\right|^2 \left|h(j\omega)\right|^2} \quad (19)$$

The neural network is trained to classify four possible faults, namely a fault in the actuator, a fault in the level sensor, a fault in the flow sensor, and a leakage.

The fuzzy approach is then integrated sequentially with the neural network-based fault classification approach to complete the required fault isolation scheme. The Neural Network-based classifier precedes the Fuzzy Logic-based one, with the former providing a fast fault classification, followed by a fuzzy logic block to unravel the real cause(s) of the fault. The fault magnitude is estimated from the changes in the settling time, $\Delta t_s = t_{ss}^0 - t_{ss}$, whereas its onset is indicated by the changes in the height profile.

Figs 6-8 give the profiles of the flow, height and the coherence spectrum. Fig. 6 shows height profiles in the presence of leakages of different magnitudes occurring when the fluid level system is operated in both an open-loop and a closed-loop configuration. For the open-loop case, one can readily deduce both the onset and amount of the leakage from the height/flow profile. The leakage flow has five sections corresponding to the following five degrees of no-leakage, small, medium, large and very large leakage. However, by its very nature, the closed-loop PI controller hides the fault and hence makes it difficult to visually detect it.

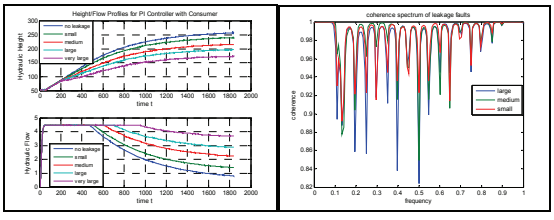


Figure 6: Height/Flow Profile/Coherence under leakage Faults.

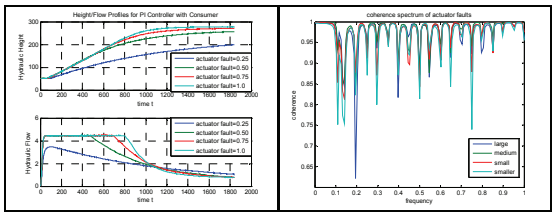


Figure 7: Height/FlowProfile/Coherence under actuator faults.

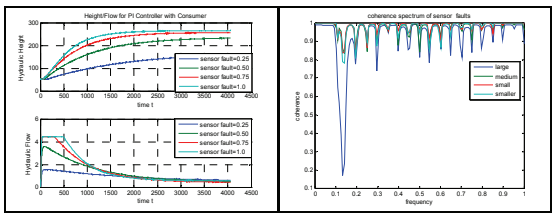


Figure 8: Height/Flow Profile/Coherence under level sensor faults.

6.2 Model of the Fluid System

A benchmark model of a cascade connection of a dc motor and a pump relating the input to the motor, *u,* and the flow, Q_i, is a first-order system expressed by:

$$\dot{Q}_i = -a_m Q_i + b_m \phi(u) \quad (20)$$

where a_m and b_m are the parameters of the motor-pump system and $\phi(u)$ is a dead-band and saturation-type nonlinearity. The Proportional and Integral (PI) controller is given by:

$$\begin{aligned} \dot{x}_3 &= e = r - h_2 \\ u &= k_p e + k_I x_3 \end{aligned} \quad (21)$$

where k_p and k_I are gains and r is the reference input.

With the inclusion of the leakage, the liquid level system is modelled by (Astrom et al., 2001):

$$A_1 \frac{dH_1}{dt} = Q_i - C_{12}\varphi\left(H_1 - H_2\right) - C_\ell \varphi\left(H_1\right) \quad (22)$$

$$A_2 \frac{dH_2}{dt} = C_{12}\varphi\left(H_1 - H_2\right) - C_0 \varphi\left(H_2\right) \quad (23)$$

where $\varphi(.) = sign(.)\sqrt{2g(.)}$, $Q_\ell = C_\ell \varphi\left(H_1\right)$ is the leakage flow rate, $Q_0 = C_0 \varphi\left(H_2\right)$ is the output flow rate, H_1 and H_2 are the liquid heights in tanks 1 and 2, respectively, A_1 and A_2 are the cross-sectional areas of tanks 1 and 2, respectively, g=980 $cm/\sec^2$ is the gravitational constant, C_{12} and C_o are the

discharge coefficients of the inter-tank and output valves, respectively. The linearized model of the entire system formed of the motor, pump, and the tanks is given by:

$$\dot{x} = Ax + Br \quad y = Cx \tag{24}$$

$$x = \begin{bmatrix} h_1 \\ h_2 \\ x_3 \\ q_i \end{bmatrix}, A = \begin{bmatrix} -a_1 - \alpha & a_1 & 0 & b_1 \\ a_2 & -a_2 - \beta & 0 & 0 \\ -1 & 0 & 0 & 0 \\ -b_m k_p & 0 & b_m k_I & -a_m \end{bmatrix}, \tag{25}$$

$$B = \begin{bmatrix} 0 & 0 & 1 & b_m k_p \end{bmatrix}^T, C = [1 \quad 0 \quad 0 \quad 0]$$

Where q_i, q_ℓ, q_0, h_1 and h_2 are respectively the increments in Q_i, Q_ℓ, Q_o, H_1^0 and H_2^0, a_1, a_2, α and β are parameters associated with linearization, α is associated with leakage and β is the output flow rate, $q_\ell = \alpha h_1$, $q_o = \beta h_2$.

6.3 Evaluation of the Fault Detection using a Bank of Kalman Filters

A bank of two Kalman filters is used here, one with input u(k) and the flow-sensor output, and the other with input u(k) and the height-sensor output

First the fault-free model of the system is identified using a recursive least-squares identification scheme. The order of the estimated model was iterated to obtain an acceptable model structure using a combination of the AIC criterion and the identified pole locations.

The identified model is essentially a second-order system with a delay even though the theoretical model is of a fourth order. Using the fault-free model together with the covariance of the measurement noise, R, and the plant noise covariance, Q, the Kalman filter model was finally derived. As it is difficult to obtain an estimate of the plant covariance, Q, a number of experiments were performed under different plant scenarios to tune the Kalman gain, K_0.

$$\hat{x}_i(k+1) = A_i\hat{x}_i(k) + B_i u_i(k-d) + K_i\left(y_i(k) - C_i\hat{x}_i(k)\right) \tag{26}$$

$$r_i(k) = y_i(k) - C_i\hat{x}(k) \qquad i = 1,2 \tag{27}$$

where x_i is the state, r_i is the residual, (A_i, B_i, C_i) is the state-space model of the first subsystem relating the control input $u(k)$ to the flow output $y_i(k)$. The transfer function for the first subsystem (A_1, B_1, C_1) relating the control input $u(k)$ to the flow output $y_1(k)$.

$$y_1(z) = G_0(z)G_1(z)u(z) \tag{28}$$

where G_0 is the actuator transfer function and G_1 is the transfer function relating the actuator output to the flow. (A_2, B_2, C_2) is the state-space model for the second subsystem relating the control input $u(k)$ to the height $y_2(k)$. The transfer function for the second subsystem (A_2, B_2, C_2) relating the control input $u(k)$ to the height output $y_2(k)$

$$y_2(z) = G_0(z)G_1(z)G_2(z)\,u(z) \tag{29}$$

where G_2 is the transfer function relating the flow to the height.

In this case, four possible fuzzy rules can be derived, two of which are stated in the following:

- If $\left|\frac{1}{N}\sum_{i=1}^{N} r_1(i)\right| > \sigma_{thr}$, then there is a fault in G_0 (subsystem 0) or G_1 (subsystem 1) or in the flow-sensor,
- If $\left|\frac{1}{N}\sum_{i=1}^{N} r_2(i)\right| > \sigma_{thr}$, then there is a fault in G_0 (subsystem 0) or G_1 (subsystem 1) or G_2 (subsystem 2) or in the height-sensor (level-sensor).

The Kalman filter bank was evaluated under different fault scenarios for an ON-OFF controller, a P controller, and a PI controller, as shown in Fig.9.

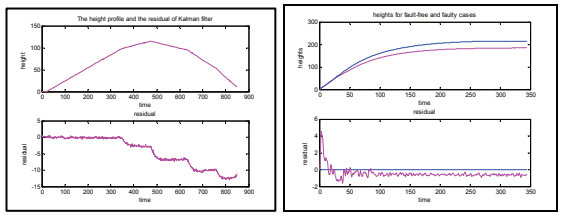


Figure 9: Kalman filter results for an On-Off and PI Controller: for Flow and Height under various leakage magnitudes.

Comments: The model of the fluid system is nonlinear, complex and stochastic. A simplified linearized model which contains only the dominant poles (as it was difficult to identify the fast dynamics) was used in the design of the Kalman filter bank. Results from the evaluation on the physical system shows that the Kalman filter bank is robust in modelling uncertainties including nonlinearities and neglected fast dynamics, while at the same time being sensitive to incipient faults.

7 CONCLUSIONS

The proposed intelligent fault diagnostic scheme based on a sequential integration of model-free and model (Kalman)-based approach was found promising when applied to a benchmarked laboratory-scale two-tank system. The model-free approach detects a presence of a possible fault from the integration of both neural network and fuzzy logic approaches. Results from the evaluation on the physical system shows that the Kalman filter bank is robust in modeling uncertainties including nonlinearities and neglected fast dynamics, while retaining its sensitivity to incipient faults. The integration of fuzzy-logic and neural networks proved itself to be a robust way of providing a quick and reliable indication of a fault based on steady-state measurements and height profile.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the support of KFUPM and the National Science and Engineering Research Council (NSERC) of Canada, in carrying out this work.

REFERENCES

S.X. Ding, " Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools" *Springer-Verlag* 2008.

Silvio Simani, Cesare Fantuzzi and Ronald J Patton, "Model-based Fault Diagnosis using Identification Technique", *Advances in Industrial Control, Springer Verlag*, 2003.

Patton, R.J. Paul M. Frank, and Robert N. Clark, "Issues in Fault Diagnosis for Dynamic Systems", *Springer-Varlag,* 2000.

Chen, J. and Patton, "Robust Model-based Fault Diagnosis for Dynamic Systems", *Kluwer Academic Publishers*, 1999.

Janos J. Gertler, "Fault Detection and Diagnosis in Engineering Systems", *Marcel Dekker Inc*, 1998.

R. Isermann, " Fault diagnosis of Machines via parameter estimation and knowledge processing", *Automatica,* Vol. 29, No.4, pp. 825-825, 1993.

R. Doraiswami, C.P.Diduch and Jiong Tang, "A Diagnostic Model For Identifying Parametric Faults", *IFAC World Congress*, July 2008

R.Doraiswami, "Modelling and identification for fault diagnosis: a new paradigm**"** *Proceedings of the 10th International Conference on Control Applications***,** September, 2001.

Marco Ferrente and Bruno Brunone, "Pipe system diagnosis and leak detection by unsteady-state tests", *Proceedings of the 7th World Congress on Intelligent Control and Automation*, Chongking, China, June 25-27, 2008.

Zhang Sheng; Toshiyuki, A.; Shoji, H., "Gas leakage detection system using Kalman filter", *7th International Conference on Signal Processing Proceedings. ICSP '04,* Volume 3, Aug31-Sept4. 2004, pp.:2533 - 2536

Doraiswami, R.; Sevenson, M.; Diduch, C.P ,"Autonomous control systems: monitoring, diagnosis and tuning", *IEEE Transactions on Systems, Man and Cybernetics, Part A,* Volume 26, Issue 5, Sept.1996 pp. 646 – 655.

R.J. Patton, F.J. Uppal and C.J. Toribio, "Soft computing approaches to fault diagnosis for dynamic systems: A survey", *in Proceedings of the IFAC Symposium SAFEPROCESS* 2000, Budapest-Hungary; 2000.

Astrom et.,al," Control of Complex Systems", *Springer-Verlag*, 2001.

C. De Persis, "On the observability codistributions of a nonlinear system" *Systems and Control Letters,* Volume: 40, Issue: 5, August 15, 2000, pp. 297-304

H. Hammouri, P. Kabore, S.Othman, and J. Biston, J "A. Failure diagnosis and nonlinear observer. Application to a hydraulic process" *Journal of The Franklin Institute*, Volume: 339, Issue: 4-5, July - August, 2002, pp. 455-478.

K.M. Kinnaert, "Robust fault detection based on observers for bilinear systems", *Automatica*, Volume: 35, Issue: 11, November, 1999, pp. 1829-1842.

DESIGN OF A NOVEL HYBRID OPTIMIZATION ALGORITHM

Dimitris V. Koulocheris and Vasilis K. Dertimanis
Vehicles Laboratory, National Technical University of Athens, Iroon Politechniou 9, 157 80, Athens, Greece
dbkoulva@central.ntua.gr, bullit@central.ntua.gr

Keywords: Hybrid optimization, Evolution strategy, Deterministic mutation, Line–search, Trust–region, Vehicles.

Abstract: The interrelation of stochastic and deterministic optimization algorithms, as well as the exploitation of the advantages that each counterpart presents simultaneously, is studied in this paper. To this, a hybrid optimization algorithm is developed, which consists of a conventional Evolution Strategy that maintains its recombination and selection phases unaltered, while its mutation operator is replaced by well–known deterministic methods, such as line–search and/or trust–region. The alteration results in superior performance of the novel algorithm, compared to other instances of Evolutionary Algorithms, as exploited out in tests using Griewangk and Rastrigin functions. The proposed algorithm is further examined through its implementation to the structural optimization problem of a full–car suspension model, with satisfying results.

1 INTRODUCTION

Numerical optimization, either deterministic (Nocedal and Wright, 2006), or stochastic (Schwefel, 1995; Baeck, 1996), has shown to be a very powerful tool in engineering, with implementation in a very wide area of applications, including structural design (Rao, 1996; Alkhatib et al., 2004; Koulocheris et al., 2003a), system identification (Koulocheris et al., 2003c), control (Fleming and Purshouse, 2002) and fault diagnosis (Dertimanis, 2006; Chen and Patton, 1999).

The corresponding schemes that have been for a long time the subject of significant research in the field of numerical optimization, are mostly divided into two main categories, deterministic and stochastic: the former, usually build a local quadratic model of the function of interest and converge rapidly to a local stationary point, given a "*good initial guess*" for the parameter vector, while the latter perform in a wide area of the search space, since, generally, the optimization procedure is conducted in parallel. Yet, both suffer from serious drawbacks, as the deterministic methods depend drastically on the initial parameter vector provided and frequently stuck in local optima, while the stochastic ones present very slow convergence rate (Vrazopoulos, 2003). To this, the idea of combining the diverse characteristics of these two optimization categories into a hybrid algorithmic structure, follows naturally. Surprisingly, at least in the engineering research field, relative works are rather limited (Koulocheris et al., 2004), the almost exclusive use of GA (refer to Appendix A for notation) is utilized (Koh et al., 2003), while applications are scarcely ever reported (Dertimanis et al., 2003). It should be noted though, that the problem of accelerating conventional EA has been faced using different techniques, such as neural networks (Papadrakakis and Lagaros, 2002).

This paper presents a methodology of interconnecting stochastic and deterministic optimization algorithms, in a way that exploits the advantages of both of them and results into a method that shows faster convergence rate, as well as increased reliability in the search for the global optimum. Among EA, the stochastic component has been selected to be the $[\mu/\rho\ (+/,)\ \lambda]$–ES, while the deterministic one belongs to the family of quasi–Newton methods and it is currently implemented using either line–search, trust–region, or a combination of both. To this, the currently proposed version of the algorithm integrates previous ones (Koulocheris et al., 2008; Koulocheris et al., 2004; Vrazopoulos, 2003), so that a more robust and flexible scheme is developed. In order to get insight about the performance of the novel optimization method, it is tested with the Griewangk and Rastrigin functions and compared with the conventional ES (in fact its multi–membered *plus* and *comma* versions), as well as a meta version of EP (Baeck, 1996). Consequently, it is applied to the problem of optimizing the characteristics of a suspension system used in ground vehicles.

The rest of the paper is organized as follows: in Sec. 2 the novel algorithm is presented and in Sec. 3, indications of its performance are illustrated, through the evaluation by theoretical objective functions, as well as an application example, corresponding to the problem of optimizing the riding comfort of a passenger vehicle. In Sec. 4 some final remarks are given, together with suggestions for further research.

2 THE HYBRID ALGORITHM

2.1 Description

The proposed hybrid algorithm with deterministic mutation aims, as already mentioned, at interconnecting the advantages of both optimization approaches. Deterministic methods are characterized, if the optimization function is regular, by a high convergence rate and accuracy in the search for the optimum. On the other hand, EA show a low convergence rate but they can search on a significantly broader area for the global optimum.

$[\mu/\rho\ (+/,)\ \lambda,\nu]$–hES is based on the distribution of the local and the global search for the optimum and it consists of a super-positioned stochastic global search, followed by a independent deterministic procedure, which is activated under conditions in specific members of the involved population. Thus, every member of the population contributes in the global search, while single individuals perform the local search. Similar algorithmic structures, the theoretical background of which pertains to the simulation of insects societies (Monmarche et al., 2000; Rajesh et al., 2001), have been presented by (Colorni et al., 1996; Dorigo et al., 2000; Jayaraman et al., 2000).

The stochastic platform has been selected to be the ES, while the deterministic counterpart is a quasi–Newton algorithm (see Sec. 2.2). It must be noted that the selection of ES among the other instances of EA is justified via numerical experiments in non–linear parameter estimation problems (Schwefel, 1995; Baeck, 1996), which have provided significant indication that ES perform better than the other two classes of EA, namely GA and EP.

The conventional ES is based on three operators that take on the recombination, the mutation and the selection tasks. In order to maintain an adequate stochastic performance in the new algorithm, the recombination and selection tasks are retained unaltered (refer to (Beyer and Schwefel, 2002) for a brief discussion about the recombination phase), while its strong local topology performance is utilized through the substitution of the original mutation operator by a quasi–Newton one.

A very important matter that affects significantly the performance of the $[\mu/\rho\ (+/,)\ \lambda,\nu]$–ES involves the members of the population that are selected for mutation: there exist indications (Koulocheris et al., 2003b) that the reason for the poor performance of EA in non–linear multimodal functions is the loss of information through the non-privileged individuals of the population. Thus, the new deterministic mutation operator is not applied to all λ recombined individuals but only to the ν worst among the $(\mu\ (+/,)\ \lambda)$, where ν is an additional algorithm parameter. This means that a sorting procedure takes place twice in every iteration step: the first time in order to yield the ν worst individuals and the second to support the selection operator, which succeeds the new deterministic mutation operator. This modification enables the strategy to yield the corresponding local optimum for each of the selected ν worst individuals in every iteration step. The advantage is reflected in terms of increased convergence rate and reliability in the search for the global optimum, while three other alternatives were tested. In these, the deterministic mutation operator was activated by:

- every individual of the involved population,
- a number of privileged individuals, and
- a number of randomly selected individuals.

The above alternatives led to three types of problematic behavior. More specifically, the first increased the computational cost of the algorithm without the desirable effect. The second alternative led to premature convergence of the algorithm to local optima of the objective function, while the third generated unstable behavior that led to statistically low performance.

2.2 The Deterministic Mutation

As noted, quasi–Newton type methods replace the original mutation of ES. Yet, unlike earlier versions (Vrazopoulos, 2003), it is not wise to limit the operator in a line–search framework, since trust–region and mixedcombined methods have also proven to be competitive alternatives, or to enforce the exclusive use of the BFGS Hessian update, as analytical or finite–difference derivative information may, in some cases, be either available, or costless to compute. This fact leads to the optional implementation of full Newton methods, but the term quasi–Newton shall be preserved, in order to cover the majority of the problems faced in practice. Thus, in the following it is assumed that the gradient of the objective function is approximated using finite–differences, while the Hessian is

calculated using the powerful BFGS update.

The currently presented version of the $[\mu/\rho\ (+/,)\ \lambda,\nu]$–ES offers three alternatives to be used as mutation operators, which are briefly discussed in the following.

2.2.1 Line–search

A line–search algorithm is build in a simple idea: at iteration k, given a descent direction $\mathbf{p}_k$, take a step in that direction that yields an "acceptable" parameter vector, that is,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \cdot \mathbf{p}_k \tag{1}$$

for some λ_k that makes $\mathbf{x}_{k+1}$ an acceptable next iterate. Since,

$$\nabla^2 f(\mathbf{x}_k) \cdot \mathbf{p}_k = \nabla f(\mathbf{x}_k) \tag{2}$$

is utilized, the proposed mutation operator implements a cubic polynomial line–search procedure for the determination of λ_k, that satisfies both Wolfe conditions (Nocedal and Wright, 2006). It must be noted that in every iteration the full quasi–Newton step ($\lambda_k = 1$) is always tested first.

2.2.2 Trust–region

If in Eq. 1 the full quasi–Newton step is unsatisfactory, it means that the quadratic model fails to approximate the objective function in this region. Instead of calculating a search direction, trust–region methods calculate a shorter step length by solving the problem,

$$m_c(\mathbf{x}_k + \mathbf{p}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \cdot \mathbf{p} + \frac{1}{2}\mathbf{p}^T \nabla^2 f(\mathbf{x}_k) \cdot \mathbf{p} \tag{3}$$

where m_c is the quadratic model, subject to,

$$\|\mathbf{p}\| \leq \delta_k \tag{4}$$

so that,

$$(\nabla^2 f(\mathbf{x}_k) + \xi \cdot I) \cdot \mathbf{p}_k = \nabla f(\mathbf{x}_k) \tag{5}$$

for some $\xi > 0$. Trust–region mutation utilizes two alternatives for the calculation of ξ: the locally constrained optimal ("hook") step and the double dogleg step (Dennis and Schnabel, 1996).

2.2.3 Combined Trust–region / Line–search

The third alternative that the proposed algorithm offers as mutation operator, is a combined trust–region / line–search framework. To this, Eqs. 3-5 are solved approximately for the direction $\mathbf{p}_k$ and if the full quasi–Newton step does not result in a sufficient decrease of the objective function, a line–search is performed, which guarantees, under certain conditions, a lower objective function value. The corresponding algorithm is described in (Nocedal and Yuan, 1998).

2.3 Termination Criteria

The termination criteria are distinguished as local, referring to the deterministic mutation and global, referring to $[\mu/\rho\ (+/,)\ \lambda,\nu]$–ES. For the former, standard tests that are presented in detail in (Nocedal and Wright, 2006) and (Dennis and Schnabel, 1996) are utilized:

- Objective function value smaller than a specified tolerance,
- relative gradient norm less than a specified tolerance,
- relative distance between two successive iterations less than a specified tolerance,
- not a descent current direction, and
- maximum mutation operator iterations exceeded.

In addition, the proposed algorithm terminates if at least one of the following occurs:

- Absolute difference between worse and best objective function less than a specified tolerance,
- maximum function evaluations exceeded, and
- maximum iterations exceeded.

3 NUMERICAL RESULTS

3.1 Performance Evaluation

In order to assess the performance of the proposed algorithm, a number $N = 100$ of independent tests were utilized using the Griewangk

$$f(\mathbf{x}) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{400 \cdot n} - \prod_{i=1}^{n} cos\Big(\frac{x_i}{\sqrt{i}}\Big) \tag{6}$$

and the Rastrigin

$$f(\mathbf{x}) = 10 \cdot n + \sum_{i=1}^{n} x_i^2 - 10 \cdot cos(2 \cdot \pi \cdot x_i) \tag{7}$$

functions, with $n = 50$ parameters and known minimum at $\mathbf{x}_m = \mathbf{0}$, $f(\mathbf{x}_m) = 0$. For the tests, a version of the algorithm with $\mu = 15$ parents and $\lambda = 100$ offspring was used, while the recombination type was panmictic intermediate with $\rho = 2$ parents for the generation of each offspring. For the mutation, the trust–region approach (using the double dogleg step) of the relative operator was implemented and in every iteration the $\nu = 3$ worse vectors were mutated. The selection was made among all the involved population (that is both parents and offspring, choice that is denoted by the $+$ sign of the full notation).

Table 1: Statistical results of the compared methods: Griewangk's function.

				Termination Reason (%)		
Method	$\overline{P}$	P_{min}	P_{max}	Convergence	Max. Iterations	Mean CPU time (s)
$(15+100,3)$–hES	-9.90	-10.06	-9.52	100	0	10
$(15+100)$–ES	-0.10	-0.14	-0.06	0	100	28
$(15,100)$–ES	-2.07	-2.83	-1.09	0	100	28
meta–EP	-0.12	-0.18	-0.09	0	100	25

Table 2: Statistical results of the compared methods: Rastrigin's function.

				Termination Reason (%)		
Method	$\overline{P}$	P_{min}	P_{max}	Convergence	Max. Iterations	Mean CPU time (s)
$(15+100,3)$–hES	1.14	0.00	2.05	100	0	5
$(15+100)$–ES	2.63	2.55	2.69	0	100	26
$(15,100)$–ES	2.56	2.35	2.66	0	100	26
meta–EP	2.59	2.52	2.62	0	100	24

Regarding the comparisons, two similar instances of the conventional ES were used, that is the $(15+100)$–ES and the $(15,100)$–ES with panmictic recombination, while a version of the meta–EP with 100 population members and 10 random members for comparison was activated. Taking under consideration the possibility of a large spectrum of orders in the final objective function value, the following quantity was formulated,

$$P_j = log_{10}(f_{final}), \quad j = 1,...,100 \tag{8}$$

and three statistics qualified the results, that is the mean, the minimum and the maximum values of the P_j's, out of the set of all the independent tests. I must be noted in every iteration, that prior to the execution of every corresponding code, the random number generator was reset, in order to initialize all the compared algorithms from the same population. As far as the termination criteria are concerned, the tolerance for the convergence of the population and the number of iterations were set equal to $macheps^{1/3}$, where *macheps* the computer precision, and 100, respectively.

The results are illustrated in Tabs.1–2, where it is clear that the hybrid algorithm has outperformed all other EA. Indeed, the $(15+100,3)$–hES with a trust–region mutation returned the best statistics among the four, while it converged in all the independent tests. On the contrary, the EA didn't managed to converge within the specified number of iterations and required $2.5-5$ times more CPU time in order to execute. Yet, in the Rastrigin function the hybrid algorithm showed premature convergence, an issue that requires further investigation. In any case, the above resulted provide significant indication about the performance of the novel algorithm and enforce its application to engineering structural problems, as the one presented next.

3.2 Application

The hybrid algorithm algorithm was subsequently applied to the problem of optimizing the performance of a passenger vehicle, in order to improve the ride comfort, under a vibration environment that generated vehicle–road interaction forces with certain spectral characteristics, corresponding to the Draft–ISO formulation (Cebon, 2000). To this, an equivalent linear full–car model with seven degrees of freedom was utilized, which is presented in Figs. 1(a)–1(b). The objective was the optimization of the suspension system under explicit structural and geometric constraints. Since a vibration environment was of interest, the root–mean–square value of the vertical acceleration,

$$f(\mathbf{x}) = \frac{1}{T}\int_0^T \ddot{x}_M^2(t)dt \tag{9}$$

was selected as objective function, subject to the following constraints:

1. Parameter Bounds:

$$1000 \leq k_{ij}^s \leq 50000 \quad (N/m) \tag{10}$$

$$100 \leq c_{ij}^s \leq 5000 \quad (N{\cdot}s/m) \tag{11}$$

for $i = f,r$ and $j = l,r$.

2. Geometry:

$$|x_M(t) + L_k{\cdot}\theta_M(t) + B_k{\cdot}\phi_M(t) - x_{ij}(t)| \leq 0.100m \tag{12}$$

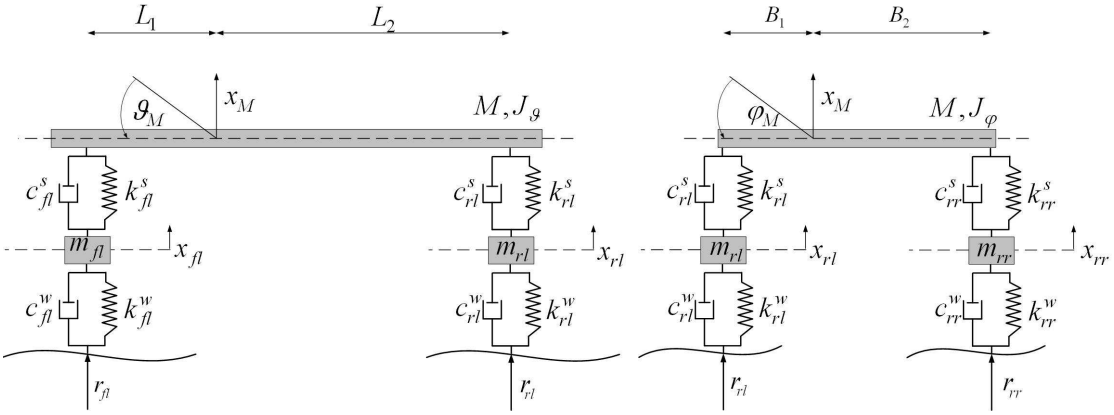


Figure 1: Structural model of a passenger vehicle: (a) pitch–bounce view and (b) roll–bounce view.

$$|x_{ij}(t)| - r_{ij}(t)| \leq 0.075m \tag{13}$$

for $k = 1,2$, $i = f,r$ and $j = l,r$.

It can be proved (see (Rao, 1996) for details) that a constrained optimization problem with low and high bounds for the involved parameters can be transformed into an unconstrained one, by applying a simple change of variable, a procedure that followed here, so that the penalty functions that were added in Eq. 9, concerned only the second type of constraints.

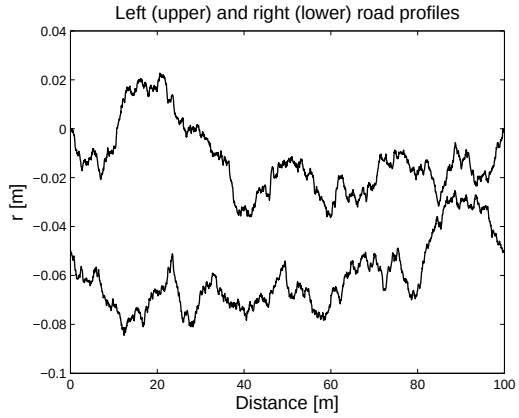


Figure 2: Two tracks across an "average" isotropic surface.

Algorithms's performance characteristics were examined via 50 Monte–Carlo experiments, each one consisting of a certain profile realization (see Fig. 2 for a single realization of the road surface topography) and 20 independent tests, for the same version of the algorithm as before, that is the $(15+100,3)$–hES.

The results are displayed in Figs. 3–4. Figure 3 displays the performance of the objective function in

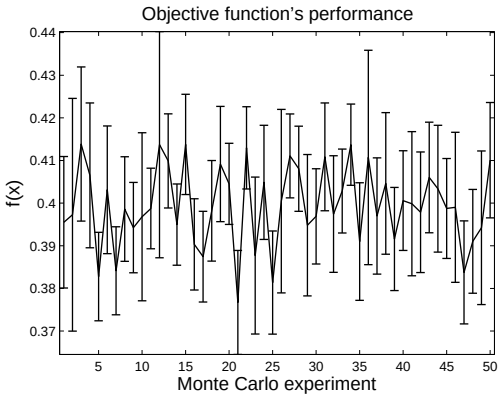


Figure 3: Mean value and dispersions of the objective function, with respect to the Monte Carlo experiments.

every Monte Carlo experiment. The horizontal line refers to the mean value of the 20 independent tests, while the vertical lines to the standard deviation of the 20 values of every Monte–Carlo experiment. It appears that the hybrid algorithm presented high statistical consistency, fact that is further supported by the suspension results that are illustrated in Fig. 4, from which clear suggestions about the front/rear suspension set up can be made. Yet, the relatively high standard deviations of the suspensions' stiffness indicate that more intuition is required about the role of these structural parameters to the root–mean–square acceleration, with respect to the mathematical model.

4 CONCLUSIONS

A novel hybrid optimization method was presented in this paper, which attempts to combine the diverse characteristics of deterministic and stochastic opti-

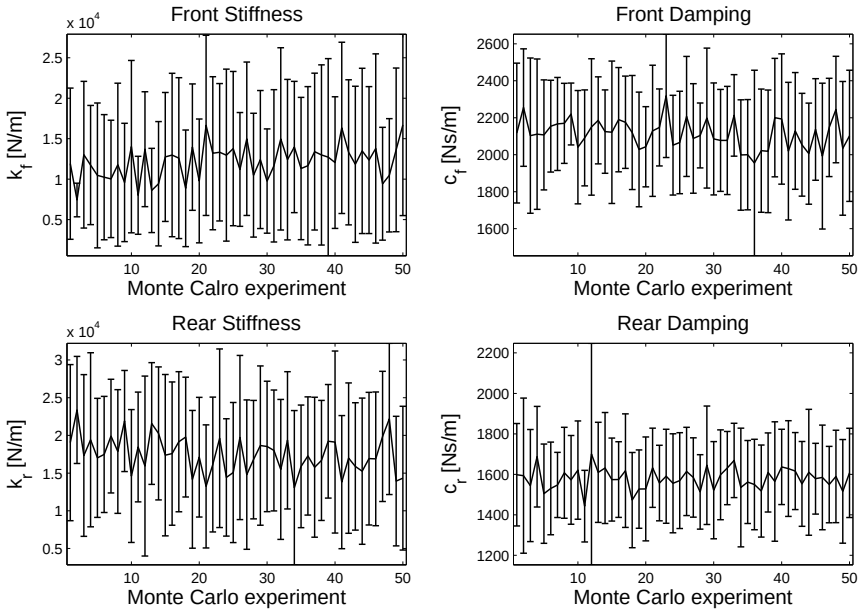


Figure 4: Mean value and dispersions of the parameter vector, with respect to the Monte Carlo experiments.

mization algorithms. That is, to interconnect fast local convergence and increased reliability in the search of the global optimum, without depending on initial values, or suffer from low convergence rate. To this, the corresponding scheme that was developed maintains the stochastic kernel of ES and replaces the original mutation operator by relative methods that utilize derivative information and act on the non-privileged population members, resulting in a more efficient performance.

The proposed algorithm was compared to conventional instances of EA using standard test functions, such as the Griewangk and Rastrigin ones, showing significant evidence about its performance, and subsequently was applied to the problem of optimizing the performance of a passenger vehicle with satisfying results that suggest, not only its use in other engineering problems, but also further investigation about its design parameters, as well as the user–supplied controls.

REFERENCES

Alkhatib, R., Nakhaie Jazarb, G., and Golnaraghi, M. (2004). Optimal design of passive linear suspension using genetic algorithm. *Journal of Sound and Vibration*, 275(3–5):665–691.

Baeck, T. (1996). *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, New York.

Beyer, H. and Schwefel, H. (2002). Evolution Strategies: a comprehensive introduction. *Natural Computing*, 1(1):3–52.

Cebon, D. (2000). *Handbook of Vehicle–Road Interaction*. Swets & Zeitlinger, Lisse.

Chen, J. and Patton, R. (1999). *Robust model–based fault diagnosis of dynamic systems*. Kluwer Academic Publishers, Massachusetts.

Colorni, A., Dorigo, M., Maffioli, F., Maniezzo, V., Righini, G., and Trubian, M. (1996). Heuristics from nature for hard combinatorial optimization problems. *International Transactions on Operational Research*, 3(1):1–21.

Dennis, J. and Schnabel, R. (1996). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Society for Industrial and Applied Mathematics, Philadelphia.

Dertimanis, V. (2006). *Fault Modeling and Identification in Mechanical Systems*. PhD thesis, School of Mechanical Engineering, GR 157 80 Zografou, Athens, Greece. In greek.

Dertimanis, V., Koulocheris, D., Vrazopoulos, H., and Kanarachos, A. (2003). Time–series parametric modeling using Evolution Strategy with deterministic mutation operators. In *Proceedings of the International Conference on Intelligent Control Systems and Signal Processing*, pages 328–333, Faro, Portugal. IFAC.

Dorigo, M., Bonabeau, E., and Theraulaz, G. (2000). Ant algorithms and stigmergy. *Future Generation Computer Systems*, 16(8):851–871.

Fleming, P. and Purshouse, R. (2002). Evolutionary algorithms in control systems engineering: a survey. *Control Engineering Practice*, 10(11):1223–1241.

Jayaraman, V., Kulkarni, B., Karale, S., and Shelokar, P. (2000). Ant colony framework for optimal design and scheduling of batch plants. *Computers and Chemical Engineering*, 24(8):1901–1912.

Koh, C., Chen, Y., and Liaw, C.-Y. (2003). A hybrid computational strategy for identification of structural parameters. *Computers & Structures*, 81(2):107–117.

Koulocheris, D., Dertimanis, V., and Spentzas, C. (2008). Parametric identification of vehicle structural characteristics. *Forschung im Ingenieurwesen*, 72(1):39–51.

Koulocheris, D., Vrazopoulos, H., and Dertimanis, V. (2003a). Design of an I-Beam using a deterministic Multiobjective optimization algorithm. *WSEAS Transactions On Circuits & Systems*, 2(4):794–799.

Koulocheris, D., Vrazopoulos, H., and Dertimanis, V. (2003b). Hybrid Evolution Strategy for the design of welded beams. In *Proceedings of EUROGEN*, Barcelona, Spain.

Koulocheris, D., Vrazopoulos, H., and Dertimanis, V. (2003c). Vehicle Suspension system identification using Evolutionary Algorithms. In *Proceedings of EUROGEN*, Barcelona, Spain.

Koulocheris, D., Vrazopoulos, H., and Dertimanis, V. (2004). A Hybrid Evolution Strategy for vehicle suspension optimization. *WSEAS Transactions On Systems*, 3(1):90–95.

Monmarche, N., Venturini, G., and Slimane, M. (2000). On how Pachycondyla apicalis ants suggest a new search algorithm. *Future Generation Computer Systems*, 16(8):937–946.

Nocedal, J. and Wright, S. (2006). *Numerical Optimization*. Springer–Verlag, New York, 2 *nd* edition.

Nocedal, J. and Yuan, Y. (1998). Combining trust region and line search techniques. In Yan, Y., editor, *Advances in Nonlinear Programming*, pages 153–175. Kluwer Academic Publishers.

Papadrakakis, M. and Lagaros, N. (2002). Reliability–based structural optimization using neural networks and Monte Carlo simulation. *Computer methods in applied mechanics and engineering*, 191(32):3491–3507.

Rajesh, J., Gupta, S., Rangaiah, G., and Ray, A. (2001). Multi–objective optimization of industrial hydrogen plants. *Chemical Engineering Science*, 56(3):999–1010.

Rao, S. (1996). *Engineering Optimization: Theory and Practice*. John Wiley & Sons Ltd., New York, 3 *rd* edition.

Schwefel, H. (1995). *Evolution & Optimum Seeking*. John Wiley & Sons Ltd., New York.

Vrazopoulos, H. (2003). *New Optimization Methods of Mechanical Systems*. PhD thesis, School of Mechanical Engineering, GR 157 80 Zografou, Athens, Greece. in greek.

APPENDIX A: NOTATION

μ	number of parent population
ρ	number of recombination population
λ	number of offspring
ν	number of mutation population
$(+/,)$	plus / comma version of ES
EA	Evolutionary Algorithms
ES	Evolution Strategy
EP	Evolutionary Pogramming
GA	Genetic Algorithms
hES	hybrid Evolution Strategy

OBSERVER-BASED STATE FEEDBACK REMOTE CONTROL WITH BOUNDED TIME-VARYING DELAYS

Imane Dilaneh and Laurent Laval
ECS–ENSEA
6, avenue du Ponceau, 95014 Cergy-Pontoise Cedex, France
Imane.Dilaneh@ensea.fr, laurent.laval@ensea.fr

Keywords: Networked Control Systems, Time-varying delays, Lyapunov–Krasovskii functional, LMI.

Abstract: This paper investigates the problem of remote stabilization via communication networks with uncertain, "non-small", time-varying, non-symmetric transmission delays affecting both the control input and the measured output. More precisely, this paper focuses on a closed-loop Master-Slave setup with a TCP network as communication media, and an observer-based state-feedback control approach to deal with the stabilization objective. First, we establish some asymptotic stability criteria regarding to a Lyapunov–Krasovskii functional derived from a descriptor model transformation, in case of "non-small" delays (that are time-varying delays with non-zero lower bounds). Then, some stability conditions are given in terms of Linear Matrix Inequalities which are used, afterwards, to design the observer and controller gains. Finally, the proposed stabilizing approach is illustrated through numerical and simulation results, related to the remote control of a "ball and beam" system.

1 INTRODUCTION

Over the past few years, the widespread development of low-cost wired and wireless data networks has lead to an increasing interest for Networked Control Systems (NCSs) (for instance, see (Yang, 2006; Tang and Yu, 2007; Hespanha et al., 2007) and references therein). Indeed, such networks seem to be suitable for large scale control systems with sensors, actuators and controllers that communicate over a shared medium. However, most of common network physical configurations and communication protocols[1] lead to transmission delays and even data losses. Then, from a control viewpoint, it is well-known that such undesirable features affect the overall NCS behavior, leading possibly to poor performance and/or instabilities (e.g. (Niculescu, 2001; Ge et al., 2007)). This justifies the increasing investigations on control strategies to insure both closed-loop stability and good performance for time-delayed systems (see (Tipsuwan and Chow, 2003; Richard, 2004) and references therein). Following this, the present paper then deals with the stabilization of a Networked Control System with consideration of TCP (Transmission Control Protocol) networking protocol for bi-directional communications between a Master system (computing the control) and a Slave system (to be controlled). In particular, we investigate the design of an observer-based (static) state-feedback controller (located in the Master system) so as to insure the asymptotic stability of the closed-loop NCS whatever the presence of time-varying, non-symmetric delays in the control and feedback loops. In this purpose, first, we establish some stability conditions by means of a Lyapunov–Krasovskii functional derived from a descriptor model transformation (Fridman and Shaked, 2002). These conditions are given in terms of Linear Matrix Inequalities which are used afterwards to design both controller and observer gains, by means of LMI optimization. This design approach is then illustrated through an example related to the remote control of a "ball and beam" system.

This paper is organized as follows. Section 2 describes the Networked Control System under consideration. Section 3 defines the observer-based control law, while section 4 focuses on the design of both state-feedback controller and observer gains. Section 5 presents a "ball and beam" system as remote controlled plant for illustrating the proposed control strategy. Then, some numerical and simulations results related to the observer-based control of this system are presented. Finally, some concluding remarks are given in section 6.

[1] Such as User Datagram Protocol (UDP), Transfer Control Protocol (TCP), Medium Access Control protocols, etc.

2 SYSTEM DESCRIPTION

Regarding to Figure 1, the Networked Control System under consideration consists in a Master-Slave setup, with a TCP network as communication media linking these two systems.

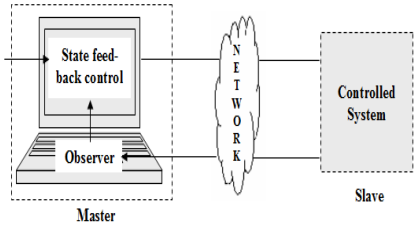


Figure 1: The Networked Control System (Master-Slave configuration).

- The exchanged data correspond respectively to the control input (sent by the Master to the Slave), and a measured output of the remote system (sent by the Slave to the Master). Due to the networking protocol and communication lines properties, we consider some time-delays τ_1 and τ_2, respectively related to the Master-to-Slave and Slave-to-Master transmissions. Moreover these delays are assumed to be time-varying, uncertain (with known lower and upper bounds), and non-symmetric (that is $\tau_1 \neq \tau_2$).

Remark 1. *The consideration of TCP networking protocol insure that all transmitted data are received in the emission order. Thus, when considering a first data packet emitted at time t_1 undergoing a delay τ_1, and a second data packet emitted at time t_2 undergoing a delay τ_2, the correct scheduling of data implies that (see (Witrant et al., 2003)):*

$$t_1+\tau_1 < t_2+\tau_2 \quad \Leftrightarrow \quad -1 < \frac{\tau_2-\tau_1}{t_2-t_1} \simeq \frac{d\tau}{dt} \qquad (1)$$

Therefore, the Master-to-Slave and Slave-to-Master delays $\tau_i(t)$ (with $i=1,2$) can be expressed as differentiable functions, and such that:

$$\forall t \geq 0, \quad \tau_i(t) = h_i + \eta_i(t),$$
$$with \quad 0 \leq \eta_i(t) \leq \mu_i, \quad \dot{\eta}_i(t) \leq d_i < 1 \qquad (2)$$

where the $\tau_i(t)$ (with $i=1,2$) are considered as time-varying bounded delays with non-zero lower bounds $h_i > 0$ (sometimes referred to as "non-small delays"). $\eta_i(t)$ is a differentiable function which characterizes a (bounded) time-varying perturbation with bounded time-derivative $\dot{\eta}_i(t) < 1$ (so that $\tau_i(t)$ are commonly referred to as slowly-varying delays – e.g. (Shustin and Fridman, 2007)), and μ_i and d_i are strictly positive, constant upper-bounds (see (Fridman, 2004)).

Moreover, we can define $\tau_i^* = h_i + \mu_i$ as an upper-bound for $\tau_i(t)$, leading finally to $h_i \leq \tau_i(t) \leq \tau_i^*$.

Remark 2. *Such an assumption on non-zero lower bounds h_i of delays is realistic. Indeed, zero or close to zero delays (corresponding to instantaneous or quasi-instantaneous transmissions) are usually not met in most of real networks (due to, at least, propagation phenomena).*

- The controlled system (within the Slave part), is supposed to be linear, controllable and observable, with a known state-space representation $(A;B;C)$. By taking into account the time-delay τ_1 (intrinsic to the Master-to-Slave transmission), this Slave system is then given by:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t-\tau_1(t)) \\ y(t) &= Cx(t) \end{aligned} \qquad (3)$$

  where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the delayed control input with an input time-delay $\tau_1(t) > 0$ that we assume to be a differentiable function satisfying to relation (2). $y(t) \in \mathbb{R}^p$ is the system output, and A, B and C are constants matrices of appropriate dimensions.

- The Master system includes an observer which aims at providing an estimation $\hat{x}(t)$ of the full state-vector $x(t)$ of the Slave system, from the output $y(t)$ it receives after a delay $\tau_2(t)$ (assuming this delay also satisfies to relation (2)). From this estimation $\hat{x}(t)$, the Master then computes the control and forwards it to the Slave.

3 THE OBSERVER-BASED STATE-FEEDBACK CONTROL

3.1 The Full-state Observer

As already mentioned, this paper considers, from the Master system viewpoint, a full-state reconstruction of the Slave state-vector $x(t)$ from the transmitted, delayed, scalar output $y(t-\tau_2)$ coming from the Slave system. As this last system is assumed to be linear, we propose here to perform this full-state estimation, by means of a Luenberger-type observer (Luenberger, 1971).

With respect to the NCS setup, the observer can

then be defined by:

$$\begin{aligned}\dot{\hat{x}}(t) &= A\hat{x}(t)+Bu(t-\tau_1(t)) \\ &\quad -L\left[y(t-\tau_2(t))-\hat{y}(t-\tau_2(t))\right] \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \quad (4)$$

where L is the observer gain which has to be designed so as to ensure a sufficiently fast convergence of $\hat{x}(t)$ towards the true system state $x(t)$, regardless of time-varying delay $\tau_2(t)$.

Remark 3. *Delay $\tau_2(t)$ is supposed to be time-varying and uncertain. Nevertheless, we assume the knowledge of an upper-bound $\tau_2^* = h_2 + \mu_2 \geq \tau_2(t)$.*

3.2 The Control Law

Regarding to the literature, many control strategies have been proposed to deal with the stabilization problem of NCS with delays. In our case, as the Luenberger-type observer is supposed to provide a full-state reconstruction $\hat{x}(t)$ of the Slave state-vector $x(t)$, we propose to investigate the use of a simple state-feedback control $u(t)$ of the following form:

$$u(t) = K\hat{x}(t) \quad (5)$$

where K is the control gain to design so as to guarantee the closed-loop stability of the controlled system (the Slave), regardless of the control input time-varying delay $\tau_1(t)$.

4 DESIGN OF CONTROLLER AND OBSERVER GAINS

With respect to (4) and (5), this section is devoted to the design of both controller and observer gains that guaranty the closed-loop stabilization of the NCS (despite of both input and output time-varying delays $\tau_1(t)$ and $\tau_2(t)$). In this aim, let us establish some asymptotic stability criteria, by applying a Lyapunov-Krasovskii methodology based on a descriptor model transformation (see (Fridman and Shaked, 2002)).

4.1 Control Design

First, let us focus on the design of an ideal controller $u(t) = Kx(t)$ by considering a perfect observer (such that $\hat{x}(t) = x(t)$), before to deal, in a later subsection, with the influence of the observation error on the whole system stability.

Thus, first, let us recall that the controlled system is represented by a linear system with bounded, time-varying, input delay (see Remark 1), whose dynamics can be expressed as:

$$\begin{aligned}\dot{x}(t) &= Ax(t)+BKx(t-\tau_1(t)) \\ x(\theta) &= \varphi(\theta), \quad \theta\in[-\tau_1^*,0] \end{aligned} \quad (6)$$

where $\tau_1^* = h_1 + \mu_1$ is an upper-bound for time-delay $\tau_1(t)$. Then, following a similar approach as in (Fridman, 2004), let us express a result that gives some asymptotic stability conditions for system (6), in terms of Linear Matrix Inequalities, for a given K.

Theorem 1. *Given a gain matrix K, system (6) is asymptotically stable if there exists $n\times n$ matrices $0 < P_1, P_2, P_3, S_1, S_{a_1}, Y_{1i}, Y_{a_{1i}}, Z_{1k}, Z_{a_{1k}}$, and R_1, R_{a_1} satisfying the LMI conditions for $i = 1,2$ and $k = 1,2,3$:*

$$\Gamma = \begin{bmatrix} \Psi_1 & P^T\begin{bmatrix}0\\BK\end{bmatrix} - Y_{a_1}^T & Y_{a_1}^T - Y_1^T \\ * & -(1-d_1)S_{a_1} & 0 \\ * & * & -S_1 \end{bmatrix} < 0 \quad (7)$$

$$\text{and,} \quad \begin{bmatrix} R_1 & Y_1 \\ * & Z_1 \end{bmatrix} \geq 0, \begin{bmatrix} R_{a_1} & Y_{a_1} \\ * & Z_{a_1} \end{bmatrix} \geq 0 \quad (8)$$

with,

$$Y_1 = [Y_{11}\ Y_{12}] \qquad Y_{a_1} = [Y_{a_{11}}\ Y_{a_{12}}]$$

$$Z_1 = \begin{bmatrix} Z_{11} & Z_{12} \\ * & Z_{13}\end{bmatrix} \quad Z_{a_1} = \begin{bmatrix} Z_{a_{11}} & Z_{a_{12}} \\ * & Z_{a_{13}}\end{bmatrix}$$

where $*$ denotes the symmetric, and Ψ_1 is given by:

$$\begin{aligned}\Psi_{11} &= P_2^TA + A^TP_2 + S_1 + h_1Z_{11} + Y_{11} + Y_{11}^T + S_{a_1} \\ &\quad + \mu_1 Z_{a_{11}} \\ \Psi_{12} &= A^TP_3 + P_1^T - P_2^T + h_1Z_{12} + Y_{12} + \mu_1 Z_{a_{12}} \\ \Psi_{13} &= -(P_3 + P_3^T) + h_1(Z_{13} + R_1) + \mu_1 R_{a_1} + \mu_1 Z_{a_{13}}\end{aligned}$$

Proof — Representing (6) in an equivalent descriptor form ((Fridman and Shaked, 2002)) leads to:

$$\begin{aligned}\dot{x}(t) &= z(t) \\ 0 &= -z(t) + Ax(t) + BKx(t-\tau_1(t))\end{aligned} \quad (9)$$

By posing $\bar{x}(t) = col\{x(t), z(t)\}$ and $E = diag\{I_n, 0\}$, then (9) can be rewritten as:

$$E\dot{\bar{x}} = \begin{bmatrix}\dot{x}(t)\\0\end{bmatrix} = \begin{bmatrix} z(t) \\ -z(t)+\Lambda x(t)\end{bmatrix} - \begin{bmatrix}0\\BK\end{bmatrix}\int_{t-h_1}^{t} z(s)ds - \begin{bmatrix}0\\BK\end{bmatrix}\int_{t-h_1-\eta_1}^{t-h_1} z(s)ds \quad (10)$$

where $\Lambda = A + BK$.

Now, considering a Lyapunov-Krasovskii functional (LKF) of the form:

$$V(t) = V_n(t) + V_a(t); \quad (11)$$

where $V_n(t)$ is a nominal LKF corresponding to the nominal system (10) with $h_1 \neq 0$ and $\eta_1(t) = 0$, and such that (see (Fridman, 2004)):

$$\begin{aligned}V_n(t) &= \bar{x}(t)^TEP\bar{x}(t) + \int_{-h_1}^{0}\int_{t+\theta}^{t} z(\beta)^TR_1z(\beta)d\beta d\theta \\ &\quad + \int_{t-h_1}^{t} x(s)^TS_1x(s)ds\end{aligned} \quad (12)$$

and $V_a(t)$ is an additional term (which corresponds to the perturbed system), which vanishes when the delay perturbation approaches to 0 (that is when $\eta_1(t) \simeq 0$) and such that:

$$\begin{aligned} V_a(t) &= \int_{-\mu_1}^{0}\int_{t+\theta-h_1}^{t} z(s)^T R_{a_1} z(s) ds d\theta \\ &\quad + \int_{t-\tau_1(t)}^{t} x(s)^T S_{a_1} x(s) ds \end{aligned} \tag{13}$$

with $P = \begin{bmatrix} P_1 & 0 \\ P_2 & P_3 \end{bmatrix}$, P_1, R_1, R_{a_1}, S_1 and $S_{a_1} > 0$.

Noting that $V_1 = \overline{x}(t)^T E P \overline{x}(t) = x(t)^T P_1 x(t)$, then differentiating this term in t along the trajectories of the perturbed system (9) leads to:

$$\frac{dV_1(t)}{dt} = 2\overline{x}(t)^T P^T \begin{bmatrix} \dot{x}(t) \\ 0 \end{bmatrix}$$

Then, replacing $[\dot{x}(t) \quad 0]^T$ by the right side of (10), the derivative of $V_n(t)$ in t along the trajectories of the perturbed system (9) satisfies the following relation:

$$\begin{aligned} \dot{V}_n(t) &= \overline{x}(t)^T \Psi_0 \overline{x}(t) + \delta_1(t) + \delta_2(t) + h_1 z(t)^T R_1 z(t) \\ &\quad - \int_{t-h_1}^{t} z(s)^T R_1 z(s) ds + x(t)^T S_1 x(t) \\ &\quad - x(t-h_1)^T S_1 x(t-h_1) \end{aligned} \tag{14}$$

with $\Psi_0 = P^T \begin{bmatrix} 0 & I_n \\ \Lambda & -I_n \end{bmatrix} + \begin{bmatrix} 0 & I_n \\ \Lambda & -I_n \end{bmatrix}^T P$.

Moreover, it comes that $\delta_1(t)$ and $\delta_2(t)$ are given by:

$$\begin{aligned} \delta_1(t) &= -2\overline{x}(t)^T P^T \begin{bmatrix} 0 \\ BK \end{bmatrix} \int_{t-h_1}^{t} z(s) ds \\ \delta_2(t) &= -2\overline{x}(t)^T P^T \begin{bmatrix} 0 \\ BK \end{bmatrix} \int_{t-h_1-\eta_1}^{t-h_1} z(s) ds \end{aligned}$$

Now, let us bound $\delta_1(t)$ and $\delta_2(t)$ by applying the bounding given in (Moon et al., 2001), where, for any $a \in \mathbb{R}^n$, $b \in \mathbb{R}^{2n}$, $R \in \mathbb{R}^{n\times n}$, $Y \in \mathbb{R}^{n\times 2n}$, $Z \in \mathbb{R}^{2n\times 2n}$, $N \in \mathbb{R}^{2n\times n}$ the following holds:

$$-2b^T N a \leq \begin{bmatrix} a \\ b \end{bmatrix}^T \begin{bmatrix} R & Y - N^T \\ Y^T - N & Z \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

with $\begin{bmatrix} R & Y \\ Y^T & Z \end{bmatrix} \geq 0$.

By considering such bounding condition and taking $N = P^T [0 \quad BK]^T$, $a = z(s)$, $b = \overline{x}(t)$, $R = R_1$, $Z = Z_1$, $Y = Y_1$ such that:

$$\begin{bmatrix} R_1 & Y_1 \\ * & Z_1 \end{bmatrix} \geq 0$$

then, we can find the following bound for δ_1 :

$$\begin{aligned} \delta_1(t) &\leq \int_{t-h_1}^{t} z(s)^T R_1 z(s) ds + h_1 \overline{x}(t)^T Z_1 \overline{x}(t) \\ &\quad + 2(x(t)^T - x(t-h_1)^T) \\ &\quad (Y_1 - \begin{bmatrix} 0 & (BK)^T \end{bmatrix} P) \overline{x}(t) \end{aligned} \tag{15}$$

Similarly to $\delta_1(t)$, by posing $N = P^T [0 \quad BK]^T$, $a = z(s)$, $b = \overline{x}(t)$, $R = R_{a_1}$, $Z = Z_{a_1}$, $Y = Y_{a_1}$, such that:

$$\begin{bmatrix} R_{a_1} & Y_{a_1} \\ * & Z_{a_1} \end{bmatrix} \geq 0$$

the bound of $\delta_2(t)$ is given by:

$$\begin{aligned} \delta_2(t) &\leq \int_{t-h_1-\eta_1}^{t-h_1} z(s)^T R_{a_1} z(s) ds + \mu_1 \overline{x}(t)^T Z_{a_1} \overline{x}(t) \\ &\quad + 2(x(t-h_1)^T - x(t-h_1-\eta_1)^T) \\ &\quad (Y_{a_1} - \begin{bmatrix} 0 & (BK)^T \end{bmatrix} P) \overline{x}(t) \end{aligned} \tag{16}$$

Then, the time-derivative of $V_a(t)$ is given by:

$$\begin{aligned} \dot{V}_a(t) &= \mu_1 z(t)^T R_{a_1} z(t) - \int_{t-h_1-\eta_1}^{t-h_1} z(s)^T R_{a1} z(s) ds \\ &\quad + x(t)^T S_{a_1} x(t) \\ &\quad - (1-d_1) x(t-h_1-\eta_1)^T S_{a_1} x(t-h_1-\eta_1) \end{aligned}$$

Substituting (15) and (16) into (14), we find that the derivative of $V(t)$ along the trajectories of the perturbed system satisfies the following inequality:

$$\dot{V}(t) \leq \zeta(t)^T \Gamma \zeta(t) \tag{17}$$

where $\zeta(t) = col\{\overline{x}(t), x(t-h_1-\eta_1), x(t-h_1)\}$, and Γ is a negative matrix given by (7). Thus $\dot{V}(t)$ is negative definite if conditions (7) and (8) are satisfied, while $V(t) \geq 0$. Therefore, system (6) is asymptotically stable, and the proof is achieved. ■

Note that conditions (7) and (8) are satisfied for a given state-feedback gain K. However, in our case (that is a stabilization problem involving a control law $u(t) = Kx(t)$), K is an unknown control gain to be designed so as to insure the closed-loop stability of system (9). In such a case, the LMI condition (7) contains a bilinear term coming from the product of the LMI variable with K, leading (7) to be a Bilinear Matrix Inequality. Therefore, to give rise to a LMI condition for computing of gain K, we can apply the transformation given in (Suplin et al., 2004). In this aim, let us define: $P_3 = \varepsilon P_2$ where $\varepsilon \in \mathbb{R}$ is a tuning scalar parameter. Moreover, let us note that P_2 is nonsingular since the only matrix which can be negative definite in the second block on the diagonal of Ψ_1 is $-\varepsilon(P_2 + P_2^T)$. Therefore we can also define: $\bar{P} = P_2^{-1}$. In addition, for any matrix $V = \{P_1, S_1, S_{a_1}, Y_{1i}, Y_{a_{1i}}, Z_{1k}, Z_{a_{1k}}, R_1, R_{a_1}\}$, for $i = 1, 2$ and

$k = 1,2,3$, let us define an other matrix $\bar{V} = \bar{P}^T V \bar{P}$. Then, by multiplying (7), from the right and the left sides respectively, by $\Delta_4 = diag\{\bar{P},\bar{P},\bar{P},\bar{P}\}$ and its transpose Δ_4^T, and multiplying (8) by $\Delta_3 = diag\{\bar{P},\bar{P},\bar{P}\}$ and its transpose Δ_3^T, from the right and the left sides respectively, and posing $W = K\bar{P}$, the proof of the following theorem is straightforward.

Theorem 2. *Suppose that, for some positive number* ε, *there exists a positive-definite matrix* $\bar{P}_1$, $n \times n$ *matrices* $\bar{P}$, $\bar{S}_1$, $\bar{S}_{a_1}, \bar{Y}_{1i}, \bar{Y}_{a_{1i}}, \bar{Z}_{1k}, \bar{Z}_{a_{1k}}$, $\bar{R}_1, \bar{R}_{a_1}$ *and* $W \in \mathbb{R}^{m\times n}$ *satisfying the LMI conditions for* $i = 1,2$ *and* $k = 1,2,3$*:*

$$\Gamma = \begin{bmatrix} \Psi_2 & \begin{bmatrix} BW \\ \varepsilon BW \end{bmatrix} - \bar{Y}_{a_1}^T & \bar{Y}_{a_1}^T - \bar{Y}_1^T \\ * & -(1-d_1)\bar{S}_{a_1} & 0 \\ * & * & -\bar{S}_1 \end{bmatrix} < 0 \quad (18)$$

$$\text{and,} \quad \begin{bmatrix} \bar{R}_1 & \bar{Y}_1 \\ * & \bar{Z}_1 \end{bmatrix} \geq 0, \quad \begin{bmatrix} \bar{R}_{a_1} & \bar{Y}_{a_1} \\ * & \bar{Z}_{a_1} \end{bmatrix} \geq 0 \quad (19)$$

where,

$$\begin{aligned} \bar{Y}_1 &= [\bar{Y}_{11}\ \bar{Y}_{12}] \qquad \bar{Y}_{a_1} = [\bar{Y}_{a_{11}}\ \bar{Y}_{a_{12}}] \\ \bar{Z}_1 &= \begin{bmatrix} \bar{Z}_{11} & \bar{Z}_{12} \\ * & \bar{Z}_{13} \end{bmatrix} \quad \bar{Z}_{a_1} = \begin{bmatrix} \bar{Z}_{a_{11}} & \bar{Z}_{a_{12}} \\ * & \bar{Z}_{a_{13}} \end{bmatrix} \end{aligned}$$

and matrix Ψ_2 *is given by:*

$$\begin{aligned} \Psi_{21} &= A\bar{P} + \bar{P}^T A^T + S_1 + h_1\bar{Z}_{11} + \bar{Y}_{11} + \bar{Y}_{11}^T \\ &\quad + \bar{S}_{a_1} + \mu_1\bar{Z}_{a_{11}} \\ \Psi_{22} &= \varepsilon\bar{P}^T A^T + \bar{P}_1 - \bar{P} + h_1\bar{Z}_{12} + \bar{Y}_{12} + \mu_1\bar{Z}_{a_{12}} \\ \Psi_{23} &= -\varepsilon(\bar{P} + \bar{P}^T) + h_1(\bar{Z}_{13} + \bar{R}_1) + \mu_1\bar{R}_{a_1} \\ &\quad + \mu_1\bar{Z}_{a_{13}} \end{aligned}$$

Then, the gain,

$$K = W\bar{P}^{-1} \quad (20)$$

asymptotically stabilizes the system (6) for delay $\tau_1(t) \leq \tau_1^*$.

4.2 Observer Design

Since the pair $(A;C)$ is assumed to be observable, it is possible to determine, in the non-delayed case (that is $\tau_2 = 0$), a gain L such that the Luenberger-type observer leads the estimation error to asymptotically converge towards zero. Now, by taking into account the variable delay $\tau_2(t)$ on the Slave output, then, from (3) and (4), the observation error $e(t) = \hat{x}(t) - x(t)$ is ruled by:

$$\dot{e}(t) = Ae(t) + LCe(t - \tau_2(t)) \quad (21)$$

We then express the following result which insures that the observer state $\hat{x}(t)$ converges sufficiently fast towards the true system state $x(t)$ despite of delay $\tau_2(t)$.

Theorem 3. *Suppose that, for some positive scalar* ε, *there exists* $n \times n$ *matrices* $0 < P_1$, P, S_2, $S_{a_2}, Y_{2i}, Y_{a_{2i}}, Z_{2k}, Z_{a_{2k}}, R_2, R_{a_2}$ *and* $X \in \mathbb{R}^{n\times p}$ *satisfying the LMI conditions for* $i = 1,2$ *and* $k = 1,2,3$*:*

$$\Gamma = \begin{bmatrix} \Psi_1 & \begin{bmatrix} XC \\ \varepsilon XC \end{bmatrix} - Y_{a_2}^T & Y_{a_2}^T - Y_2^T \\ * & -(1-d_2)S_{a_2} & 0 \\ * & * & -S_2 \end{bmatrix} < 0 \quad (22)$$

$$\text{and,} \quad \begin{bmatrix} R_2 & Y_2 \\ * & Z_2 \end{bmatrix} \geq 0, \begin{bmatrix} R_{a_2} & Y_{a_2} \\ * & Z_{a_2} \end{bmatrix} \geq 0 \quad (23)$$

where,

$$\begin{aligned} Y_2 &= [Y_{21}\ Y_{22}], \quad Y_{a_2} = [Y_{a_{21}}\ Y_{a_{22}}] \\ Z_2 &= \begin{bmatrix} Z_{21} & Z_{22} \\ * & Z_{23} \end{bmatrix}, \quad Z_{a_2} = \begin{bmatrix} Z_{a_{21}} & Z_{a_{22}} \\ * & Z_{a_{23}} \end{bmatrix} \end{aligned}$$

with matrix Ψ_1 *is given by:*

$$\begin{aligned} \Psi_{11} &= P^T A + A^T P + S_2 + h_2 Z_{21} + Y_{21} + Y_{21}^T \\ &\quad + S_{a_2} + \mu_2 Z_{a_{21}} \\ \Psi_{12} &= \varepsilon A^T P + P_1^T - P^T + h_2 Z_{22} + Y_{22} + \mu_2 Z_{a_{22}} \\ \Psi_{13} &= -\varepsilon(P + P^T) + h_2(Z_{23} + R_2) + \mu_2 R_{a_2} \\ &\quad + \mu_2 Z_{a_{23}} \end{aligned}$$

Then, the gain

$$L = (P^T)^{-1} X \quad (24)$$

leads the estimation error $e(t) = \hat{x}(t) - x(t)$ *to asymptotically converge towards zero.*

Proof — Representing (21) in an equivalent descriptor form (Fridman and Shaked, 2002):

$$E\dot{\bar{e}}(t) = \begin{bmatrix} z(t) \\ -z(t) + Ae(t) + LCe(t - \tau_2(t)) \end{bmatrix} \quad (25)$$

where $\bar{e}(t) = col\{e(t), z(t)\}$, $E = diag\{I_n, 0\}$.

Recalling that $\tau_2(t) = h_2 + \eta_2(t), 0 \leq \eta_2(t) \leq \mu_2, \dot{\eta}_2(t) \leq d_2 < 1$, the proof of this theorem use the same Lyapunov-Krasovskii as given by (11) with a single delay $\tau_2(t)$:

$$V(t) = V_n(t) + V_a(t)$$

where $V_n(t)$ is a nominal LKF corresponding to the nominal system (25) with $h_2 \neq 0$ and $\eta_2(t) = 0$, and such that (see (Fridman, 2004))

$$\begin{aligned} V_n(t) &= \bar{e}(t)^T E P \bar{e}(t) + \int_{-h_2}^{0}\int_{t+\theta}^{t} z(\beta)^T R_2 z(\beta) d\beta d\theta \\ &\quad + \int_{t-h_2}^{t} e(s)^T S_2 e(s) ds \end{aligned} \quad (26)$$

and $V_a(t)$ is an additional term of the following form:

$$\begin{aligned} V_a(t) &= \int_{-\mu_2}^{0}\int_{t+\theta-h_2}^{t} z(s)^T R_{a_2} z(s) ds d\theta \\ &\quad + \int_{t-\tau_2(t)}^{t} e(s)^T S_{a_2} e(s) ds \end{aligned} \quad (27)$$

Then, by differentiating of $V(t)$ along the trajectories of system (25), and posing $P = P_2$, $P_3 = \varepsilon P_2$, where $\varepsilon \in \mathbb{R}$ is a tuning scalar parameter, then the proof is achieved by noting that $X = P^T L$. ■

5 ILLUSTRATIVE EXAMPLE

This section aims at illustrating the theoretical results of section 4, through an example related to the remote control of a "ball and beam" system. Regarding to Figure 2, this plant mainly consists in a steel ball rolling on two parallel tensioned wires. These are mounted on a beam, pivoted at its center, such that the beam angle may be controlled by a servo-motor and sensed by transducers to provide measurements of the beam angle and ball position.

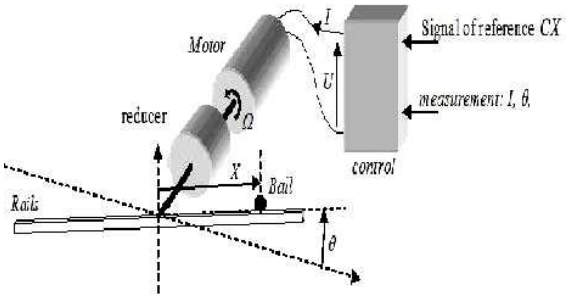


Figure 2: The Ball and Beam system to be controlled.

Regarding to the control scheme of Figure 3, the fast dynamics of the plant are regulated by two inner loops (with PI and PD controllers located in the Slave systems), so that the remaining control problem is to regulate the ball position by varying the beam angle.

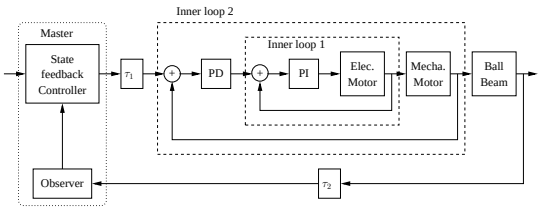


Figure 3: Control scheme of the Slave system.

According to this, the system dynamics to be controlled by means of the remote observer-based state-feedback controller, can then be defined by:

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ k_b \end{bmatrix} u(t-\tau_1(t)) \\ y(t) &= \begin{bmatrix} k_x & 0 \end{bmatrix} x(t) \end{aligned} \tag{28}$$

where $x(t) = [x_x(t)\ \ x_v(t)]^T \in \mathbb{R}^2$ is the state-vector, $x_x(t)$ and $x_v(t)$ correspond respectively to the position and the speed of the ball. $u(t-\tau_1(t))$ is the control input (with input delay $\tau_1(t)$)), $y(t)$ is the measured output (corresponding to the ball position) which is forwarded to the Master system. k_b and k_x are two constant parameters (with $k_b = 6.1\ ms^{-2}rad^{-1}$ and $k_x = 7\,V/m$).

Now, let us consider non-symmetric delays $\tau_1(t) \neq \tau_2(t)$, with, according to (2): $h_1 = 0.3s$, $h_2 = 0.25s$, $\mu_1 = \mu_2 = 0.1s$ (recalling that h_1 and h_2 are constant values, while $\eta_1(t)$ and $\eta_2(t)$ are time-varying perturbations bounded by μ_1 the and μ_2 respectively). Moreover, let us consider $d_1 = d_2 = 0.1$. By applying Theorem 2 to (6) for $\varepsilon = 9$, we find the LMI (18) is feasible for symmetric, positive-definite matrices:

$$\bar{P}_1 = \begin{bmatrix} 1.98 & 0 \\ 0 & 1.98 \end{bmatrix} \bar{R}_1 = \begin{bmatrix} 2.29 & -0.64 \\ -0.64 & 1.38 \end{bmatrix}$$

$$\bar{R}_{a_1} = \begin{bmatrix} 2.13 & -0.45 \\ -0.45 & 1.48 \end{bmatrix} \bar{S}_1 = \begin{bmatrix} 0.12 & -0.08 \\ -0.08 & 0.11 \end{bmatrix}$$

$$\bar{S}_{a_1} = \begin{bmatrix} 0.1934 & -0.1241 \\ -0.1241 & 0.5671 \end{bmatrix}$$

and,

$$\bar{P} = \begin{bmatrix} 0.5138 & -0.2327 \\ -0.2327 & 0.3694 \end{bmatrix} \quad W^T = \begin{bmatrix} -0.0088 \\ -0.0699 \end{bmatrix}$$

With respect to (20), the state-feedback controller gain K is then given by,

$$K = \begin{bmatrix} -0.1440 & -0.2800 \end{bmatrix} \tag{29}$$

Now, by applying Theorem 3 to (21) for $\varepsilon = 5.5$ (tuned by trial and error), we find the LMI (22) is feasible for symmetric, positive-definite matrices:

$$P_1 = \begin{bmatrix} 8.27 & 0 \\ 0 & 8.27 \end{bmatrix} R_2 = \begin{bmatrix} 1.05 & -1.46 \\ -1.46 & 10.29 \end{bmatrix}$$

$$R_{a_2} = \begin{bmatrix} 4.22 & -2.16 \\ -2.16 & 15.84 \end{bmatrix} S_2 = \begin{bmatrix} 0.62 & -0.23 \\ -0.23 & 0.70 \end{bmatrix}$$

$$S_{a_2} = \begin{bmatrix} 0.157 & -0.112 \\ -0.112 & 0.373 \end{bmatrix}$$

with,

$$P = \begin{bmatrix} 0.963 & -2.240 \\ -2.240 & 9.964 \end{bmatrix} \quad X = \begin{bmatrix} -0.069 \\ -0.097 \end{bmatrix}$$

Then, from (24), we finally obtain the observer gain:

$$L = [-0.198 \quad -0.054]^T \tag{30}$$

By considering the control scheme of figure 3 (meaning that the two inner loops are taking in account in simulating the dynamical behavior of the closed-loop Master-Slave system), and numerical results (29)–(30) for the controller and observer gains respectively, we then obtain the simulation results of figures 4 and 5 (for delays $h_1 = 0.3s$, $h_2 = 0.25s$). Figure (4) represents the ball position on the beam axis when dealing with a step response of the closed-loop system

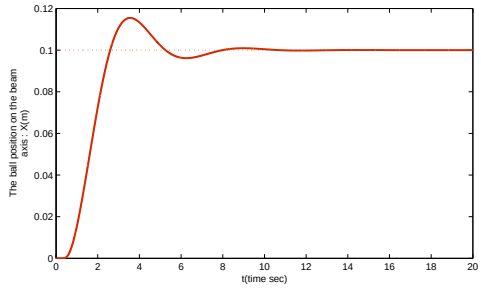


Figure 4: Step response of the closed-loop system.

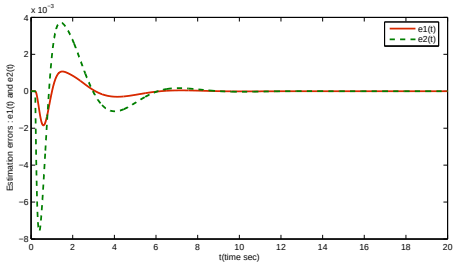


Figure 5: Estimation error $\hat{x}(t) - x(t)$.

with a step magnitude 0.1 m, while Figure (5) represents the observations errors $e_1(t) = \hat{x}_x(t) - x_x(t)$ and $e_2(t) = \hat{x}_v(t) - x_v(t)$. Moreover, Figure (6) shows the corresponding delayed control input.

By looking at these simulations results, we can see that the Luenberger-type observer insure the asymptotic convergence of the estimation error towards zero, while the state-feedback control guarantees the asymptotic stability of the closed-loop system, whatever the presence non-symmetric delays $\tau_1 \neq \tau_2$ in the control and feedback loops.

6 CONCLUSIONS

This paper has dealt with the stabilization problem of a Networked Control System with a TCP network as communication media. In particular, our attention was focusing on a Master-Slave setup with uncertain, time-varying, ”non-small”, non-symmetric transmission delays affecting the Slave control input and its transmitted (scalar) output. A main feature of our work was the use of a Lyapunov-Krasovskii functional derived from a descriptor model transformation, to give rise to some conditions for the design of an observer-based state-feedback control. In future works, we will study the stability of Networked Control Systems with both delays and packet dropping.

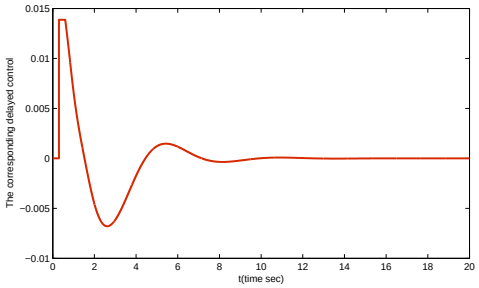


Figure 6: The corresponding delayed control input.

REFERENCES

Fridman, E. (2004). Stability of linear functional differential equations: A new lyapunov technique. In *Proc. of the Mathematical Theory of Networks and Systems*.

Fridman, E. and Shaked, U. (2002). An improved stabilization method for linear time-delay systems. *IEEE Transactions on Automatic*, 47:1931–1937.

Ge, Y., Tian, L., and Liu, Z. (2007). Survey on the stability of networked control systems. *Journal of Control Theory and Applications*, 5(4):374–379.

Hespanha, J., Naghshtabrizi, P., and Xu, Y. (2007). A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162.

Luenberger, D. (1971). An introduction to observers. *IEEE Trans. on Automatic Control*, AC-16(6):596–602.

Moon, Y., Park, P., Kwon, W., and Lee, Y. (2001). Delay-dependent robust stabilization of uncertain state delayed systems. *Int. J. of Contr*, 74:1447–1455.

Niculescu, S.-I. (2001). *Delay Effects on Stability*. LNCIS No.269, Springer.

Richard, J.-P. (2004). Time delay systems: an overview of some recent advances and open problems. *Automatica*, 39:1667–1694.

Shustin, E. and Fridman, E. (2007). On delay-derivative-dependent stability of systems with fast-varying delays. *Automatica*, 43:1649–1655.

Suplin, V., Fridman, E., and E Shaked, U. (2004). h_∞ control of linear uncertain time-delay systems- a projection approach. In *Proc. of the CDC*.

Tang, X. and Yu, J. (2007). *Networked Control System: Survey and Directions*, pages 473–481. Springer edition.

Tipsuwan, Y. and Chow, M.-Y. (2003). Control methodologies in networked control systems. *Control Engineering Practice*, 11(10):1099–1111.

Witrant, E., Canudas de Wit, C., and Alamir, M. (2003). Remote output stabilization under two channels time-varying delays. In *Proc. of the 4th IFAC Workshop on Time Delay Systems*.

Yang, T. (2006). Networked control system : A brief survey. *Control theory and applications (IEEE proceedings)*, 153(4):403–412.

RESAMPLING BASED ON STATISTICAL PROPERTIES OF DATA SETS

Julia Bondarenko
Department of Economic and Social Sciences, Helmut-Schmidt University Hamburg
(University of the Federal Armed Forces Hamburg), Holstenhofweg 85, 22043 Hamburg, Germany
bonda@hsu-hh.de

Keywords: Resampling, Classification algorithm C4.5, Uniform/(truncated) Normal distribution, Kurtosis, Chi-squared test, Kolmogorov-Smirnov test, Traffic injuries number.

Abstract: In imbalanced data sets, classes separated into majority (negative) and minority (positive) classes, are not approximately equally represented. That leads to impeding of accurate classification results. Well balanced data sets assume uniform distribution. The approach we present in the paper, is based on directed oversampling of minority class objects with simultaneous undersampling of majority class objects, to balance non-uniform data sets, and relies upon the certain statistical criteria. The resampling procedure is carried out for the daily traffic injuries data sets. The results obtained show the improving of rare cases (positive class objects) identification with accordance to several performance measures.

1 INTRODUCTION

Numerous machine learning classification methods currently give good performance in numerous practical problems, such as diagnosing medical problems, speech recognition, expert systems, robotic processing etc. A starting point for the present study was an investigation of road injuries number within the framework of the joint project with police departments of one of German federal states. The presented work focuses on temporal factors impact on daily traffic injuries number. The initial data sets of traffic injuries are imbalanced: daily injuries numbers are not approximately equally represented, that is, separated into majority (negative) and minority (positive) classes. As a result, minority class is poorly performed by classification (S. Ertekin and Giles, 2007), (S. Kotsiantis and Pintelas, 2006). But our aim is to detect efficiently the important rare cases in number of injured persons. In fact, the ability to predict periods of high incidence of road accidents, is really essential. Rebalancing the class distributions for the further classification, which includes over- and undersampling techniques, can be applied in order to solve this problem at the data level. We propose here a simple and general resampling procedure, improving a classification performance of daily road injuries number.

Oversampling method balances data set by increasing the number of minority class objects (examples). The simplest oversampling method - random oversampling - increases the minority class size by randomly replicating existing minority class examples (oversampling with replacement). This techniques is attractive exactly due to its simplicity, but unfortunately, since random oversampling only replicates existing data, it does not add any actual information to the data set. The another approach is to oversample the positive minority class by creating new examples. SMOTE (Synthetic Minority Over-Sampling Technique, see (N. Chawla and Kegelmeyer, 2002)) is the most popular oversampling method here. In SMOTE minority classes are oversampled by generating "synthetic" examples of minority class and adding them to the data set. As a result, the class distribution in the data set changes and probability of correctly classifying minority class increases. Other oversampling approaches were also proposed ((V. Garcha and Mollineda, 2008), (H. Han and Mao, 2005)). Undersampling approaches try to decrease the number of major class examples. However this method may involve information loss (that is, discard potentially important for learning and prediction examples, see (X.-Y. Liu and Zhou, 2006)).

Naturally, well balanced data sets assume uniform distribution. The approach we present in the paper, is based on directed oversampling of minority class objects with simultaneous undersampling of major-

ity class objects, to balance non-uniform data sets (to ensure the uniform distribution), and relies upon the statistical criteria. The remaining parts of the paper are organized as follows. In Section 2, we discuss the initial injuries data sets used in our work and perform their classification according to several metrics. Resampling procedure description and new classification results are presented in Section 3. Section 4 contains conclusions, brief discussion of ongoing and potential future research topics.

2 CLASSIFICATION OF INITIAL DATA SETS

2.1 Data Sets and Attributes

Data sets used in our study include the daily number of injured persons in traffic accidents in two cities of North Rhine-Westphalia, Germany - Duesseldorf and Duisburg, for the period 2004-2006. The daily data for each city represent an age group 25-59 years old people. Table 1 shows the mean, mode, median, minimum and maximum values for each region, as well as skewness and kurtosis, which are the measures of the data sets asymmetry and peakedness, respectively. Note, that we removed before all "heavy" outliers from the data sets, as sensitive to outliers. "Heavy" outliers are identified as all values more than $3IQR$ above the 3rd or below the 1st quartile, where $IQR = Q_3 - Q_1$ is interquartile range, Q_3 and Q_1 are 3rd and 1st quartiles correspondingly.

Table 1: Descriptive Statistics of Data Sets.

value	Duesseldorf	Duisbirg
mean	5.03	2.43
min	0	0
max	19	12
mode	4	1
skewness	0.65	0.92
kurtosis	3.67	4.02

Histograms showing distribution of injured persons for each urban region for period 2004-2006, are presented in Figure 1.

Let $Y_{i,t}$ be the number of injured at the day t, $t = 1,...,1096$ (3 years, 2004-2006) in region i, $i \in I = \{1,2\}$ (1=Duesseldof, 2=Duisburg). We treat as attributes the certain day features related to corresponding outcomes $Y_{i,t}$, namely:

- year: 1=2004, 2=2005, 3=2006;
- month: 1=January, 2=February,..., 12=December;

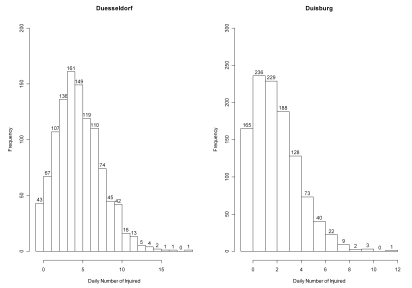


Figure 1: Histograms of Daily Number of Traffic Injured Persons for Age Range 25-59, 01.01.2004-31.12.2006.

- school vocation day: 1=yes or 0=no;
- holiday (official): 1=yes or 0=no;
- weekday: 1=Monday, 2=Tuesday,..., 7=Sunday;
- bridge day: 1=yes or 0=no.

The distribution for Region 1 looks roughly symmetric and just lightly (positively) skewed, data set for Region 2 shows more skewed behavior. Each data set has relatively narrow range of values - $0 \div 19$ for Region 1 and $0 \div 12$ for Region 2.

A number of algorithms has been developed for decision tree construction, but we will dwell upon one of them - C4.5-algorithm (Quinlan, 1993), based on the computing the metrics known as the information gain (IG) and gain ratio (GR). We introduce a notion of IG as follows:

$$IG(S_n, A) = \mathcal{E}(S_n) - \sum_{i \in values(A)} \frac{l_i}{n} \mathcal{E}(S_{A_i}) \ , \qquad (1)$$

where $\mathcal{E}(S_n) = -\sum_k \frac{n_k}{n} \log\left(\frac{n_k}{n}\right)$ is entropy of the entire data set $S_n = (Y_1,...,Y_n)$ of size n, n_k is the number of instances in S_n with value k, $\mathcal{E}(S_{A_i})$ is entropy of the sample S_{A_i} of size l_i involving elements from $\mathbf{Y}$, which correspond to the outcome (value) A_i of the attribute (feature) A, and $\frac{l_i}{n}$ represents the fraction of the data in S_n that goes into S_{A_i}.

In C4.5 algorithm below we adopt the information gain ratio to select the best day attribute to branch on at each stage. Attribute with the highest gain ratio gives us the crucial information concerning the temporal distribution of traffic injuries number inside the each region.

The formula aggregates over the different values A_i attribute A can have. But IG would be biased towards selecting attributes with more values. To mitigate this effect, we use a normalized version of IG - Gain Ratio (GR), defined as follows:

$$GR(S_n, A) = \frac{IG(S_n, A)}{Split\ Info(S_n, A)} \ , \qquad (2)$$

where split information $Split\ \ Info(S_n, A) = \sum_{i \in values(A)} \frac{l_i}{n} \ln\left(\frac{l_i}{n}\right)$ is the entropy of partitioning, or in other words, entropy associated with the distribution of the attribute A, where $\frac{l_i}{n}$ is the probability (proportion) of observing the ith value of A. Thus, a large number of small partitions is penalized there.

2.2 Evaluation Measures

In the first place, we employ the Accuracy traditional metric, that is the percentage of the correctly classified data. An information about actual and predicted examples is contained in confusion matrix (Kohavi and Provost, 1998). The entries in the confusion matrix have the following meaning in our problem:
TN is the number of correct predictions that an example is from negative class;
TP is the number of correct predictions that an example is from positive class;
FN is the number of incorrect predictions that an example is from negative class;
FP is the number of incorrect predictions that an example is from positive class.
The Accuracy is computed then as a proportion: $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$.

But for classification of imbalanced data sets, accuracy is no longer a proper measure since minority class has very small impact on the accuracy. Therefore, the following alternative evaluation measures (metrics) were proposed and used, for instance, in (G. Cohen and Geissbuhler, 2005), (Hido and Kashima, 2008), (Kubat and Matwin, 1997):

- True Positive Rate (TPR), or Recall. The proportion between correctly classified positive examples and that are calculated: Recall$= TPR = \frac{TP}{TP+FN}$.

- Precision. The proportion between correctly classified positive examples and that are actually correct: Precision$= \frac{TP}{TP+FP}$.

- G-mean (Geometric mean). Tries to maximize the accuracy on each of the two classes while keeping these accuracies balanced: $G-mean = \sqrt{Positive\ Accuracy * Negative\ Accuracy}$,
where $Positive\ \ Accuracy = \frac{TP}{TP+FN}$ =Recall, $Negative\ Accuracy = \frac{TN}{TN+FP}$.

- F-measure. "The trade-off" between precision and recall, drops rapidly if either precision or recall is poor: $FM = \frac{2\text{Recall}*\text{Precision}}{Recall+Precision}$.

2.3 Classification Results

In Section 2.1 we have noted that each data set S is kept within a certain limited relatively narrow range of values. That allows us to consider every value of the range ν_i, where $v = (v_i)^T = (\min(S), ..., \max(S))^T$, as a separate class:

Class 1: "0 Injured Persons per Day", Class 2: "1 Injured Person per Day",..., Class r "r Injured Persons per Day", where $r = 19+1 = 20$ for Region 1 and $r = 12+1 = 13$ for Region 2. Thus, we transform here count data into categorical ones. Of course, we could group the data or consider continuous data as well.

We make no assumption about the distribution of the daily number of persons injured in traffic accidents, except its unimodality during the resampling procedure. The unimodality of initial data sets is also established in the frequency distribution histograms, see Fig. 1. Remind, that a distribution is called unimodal if there is only one major "peak" (mode) in the distribution. Let M be mode of the data set, and φ_M - its frequency. We select all values of daily number of injured persons ν, those frequencies are lying within the interval from 0.8 φ_M till φ_M, to be "negative", or majority, class S^{neg}. All the other values combined together represent "positive", or minority, class S^{pos}.

The classification procedure with algorithm C4.5 gives the following results for both regions, which we present below. The classification is performed by means function J48() implemented in RWeka package, R statistical software. The minimal number of instances per leaf we set to 20. The evaluation measures defined in previous subsection, are computed either.

The classification results for both regions are reported below. Only about 18% of examples are classified correctly for Region 1, with low magnitudes of alternative measures.

```
CLASSIFICATION RESULTS FOR REGION 1:

Correctly Classified Instances 18.3394 %,

Recall = 0.1325, Precision = 0.0676,
G-mean = 0.1650, F-measure = 0.0895.
```

Majority (negative) class is presented here by numbers 3, 4 and 5. All the others are classified into the minority class. For the Region 2 we have at the beginning about 26% of correctly classified examples

```
CLASSIFICATION RESULTS FOR REGION 2:

Correctly Classified Instances 26.2774 %

Recall = 0.2449, Precision = 0.1347,
G-mean = 0.2577, F-measure = 0.1738.
```

Consistently, negative class consists of numbers 1, 2 and 3, the rest of the numbers is in positive class. In the next Section we describe a resampling strategy, which improves classification performance.

3 NEW RESAMPLING PROCEDURE

3.1 Resampling Motivation and Procedure Illustration

As one can see, in both our data sets, the examples from minority class are much less "beloved" by classification algorithm than from majority one (common problem of imbalanced data). Below we present an algorithm, which resizes/rebalances our data sets. With accordance to certain criteria, we generate artificial data from minority class and simultaneously withdraw data from majority class, until the classes are approximately equally represented (data are uniformly distributed). It may be considered as a preprocessing procedure for further classification and prediction.

The algorithm assumes unimodal character for frequency distribution of classes, without reference to skewness (asymmetry) of the distribution. Majority classes are concentrated around peak M (Figure 2a), while minority classes are more tails-sited. Our suggestion is to generate new synthetic data from singly left- and right-truncated normal (half-normal) distributions, with truncation points $u_{left} = \min(S)$ and $u_{right} = \max(S)$, respectively, where u_{left} and u_{right} are also the means of truncated normal distributions. At the same time, we flatten the peak of distribution, picking out the data placed around it. Such combination of the oversampling the minority class with undersampling the majority class helps to achieve better classifier performance.

If more detailed, a random variable is said to be from a left-truncated normal distribution if its density is

$$f(x) = \frac{q}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right)$$
$$\text{for } x \geq u_{left},\ f(x) = 0 \text{ for } x < u_{left}\ , \qquad (3)$$

where m is a mean and σ is a standard deviation of the distribution, q is a normalizing quantity with value obtained from the equation $\int_{u_{left}}^{\infty} f(x)\,dx = 1$. A right-truncated normal distribution is defined analogously. We discard elements outside the limit points u_{left} and u_{right}, and choose $m_{left} = u_{left}$ and $m_{right} = u_{right}$. This guarantees, that the least popular classes lying at the ends of data range interval, will be treated most intensive. Choosing the desired level of significance α, we put the critical points (confidence intervals endpoints with $100(1-\alpha)\%$ -confidence level) $C^{1-\alpha}_{left}$ and C^{α}_{right} for both truncated distributions equal to M. Thus, the majority class will be also maintained, as artificially generated data can be also out of endpoints $C^{1-\alpha}_{left}$ and C^{α}_{right}. The standard deviations of the truncated normal distributions can be obtained as $\sigma_{left} = \frac{M - m_{left}}{C^{1-\alpha}}$ and $\sigma_{right} = \frac{m_{right} - M}{C^{\alpha}}$, where C^{α}, $C^{1-\alpha}$ are critical values of truncated at 0 standard normal distribution. For example, for $\alpha = 0.05$: $C^{1-\alpha} = -C^{\alpha} = 1.959964$. This is shown schematically in Figure 2b.

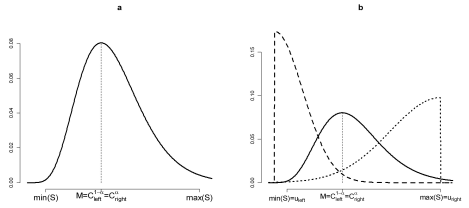


Figure 2: Scheme of Resampling Procedure: a - Underlying Unimodal Distribution; b - Underlying Distribution (solid line) and Two Truncated Normal Distributions (dashed and dotted lines).

As we have noted above, simultaneously with adding new "artificial" observation to positive class, we drop randomly observation from negative class. Therefore, we keep the sample size fixed. That helps us to avoid increasing computational time and losing information.

In the procedure, we will use the following criteria:
- Chi-squared test (Pearson goodness-of-fit test). We have data set S_n that is grouped into discrete classes. Let φ denote a vector of observed frequencies of classes and let φ^0 denote the corresponding vector of expected (specified) frequencies. We then calculate our test statistic:

$$T = \sum_{i=1}^{r} \frac{\left(\varphi_i - \varphi_i^0\right)^2}{\varphi_i}\ , \qquad (4)$$

where r is a number of classes. Under the null hypothesis, this statistic is chi-squared distributed, with degrees of freedom equal to $r-1-j$, where j is a number of parameters that should be estimated (for parametric case). That is, we can test here a null hypothesis that the frequencies of observed outcomes (daily numbers of road injuries) follow a specified (uniform) distribution, at some chosen level γ. Unfortunately, this test can be also unsuitable for samples of considerably large size (> 3000). It cannot be also applied when the expected frequency of any cell is less than 5 or the total n is less than 50. But in our case this test is appropriate, so further we apply it as a primary criterion.

- One-sample Kolmogorov-Smirnov nonparametric test (KS-test). Let $S_n = \{Y_1, Y_2, ..., Y_n\}$ be as before

our data set, with distribution function $F(y)$. We wish to test the null hypothesis H_0: $F(y) = F_0(y)$ for all y against the alternative H_1: $F(y) \neq F_0(y)$, where $F_0(y)$ is a completely specified distribution function (in our case - function of uniform distribution). Test on H_0 vs H_1 is determined by Kolmogorov-Smirnov statistic

$$D_n = \sup_{-\infty<y<\infty} |F_n(y) - F_0(y)| , \qquad (5)$$

where $F_n(y)$ is the empirical distribution function defined by $F_n(y) = \frac{1}{n}\sum_{i=1}^{n} I\{Y_i < y\}$. That is, the Kolmogorov-Smirnov test tries to determine if distribution of our data set differs significantly from the specified hypothetical distribution (here - the uniform hypothetical distribution). The null hypothesis is rejected at level γ if the computed value D_n is larger than the critical value C_γ. The critical values of Kolmogorov-Smirnov test statistic depend on the sample size. For large samples (≥ 40) one uses asymptotic critical values, which are strictly decreasing functions of the sample size: for example, the critical value at the $\gamma = 0.05$ level is approximately $\frac{1.36}{\sqrt{n}}$, where n is sample size. Obtaining the value $D_n < C_\gamma$ for considerably large n (> 3000) may dramatically increase computational time. Another problem is that Kolmogorov-Smirnov test doesn't work well with discrete (count) data. This problem could be solved, for example, by the following way: we can transform our data to continuous ones by means of Monte Carlo simulation, and then apply Kolmogorov-Smirnov test to the empirical distribution function of continuous simulated data and the specified distribution $F_0(y)$.

- As we constrain a uniform distribution, one can use a value of kurtosis for distribution control. Remind, that kurtosis is the degree of peakedness of a distribution. Removing observations from the center to the tails and "shoulders" of the distribution will decrease kurtosis, making the initial leptokurtic distributions more platykurtic. A uniform distribution has a kurtosis of 1.8. Thus, kurtosis could be used. as a simple parameter for comparison data set distribution with uniform ones: if

$$|\mu(S_n) - \mu_0| < \varepsilon , \qquad (6)$$

where $\mu_0 = 1.8$, $\mu(S_n)$ is kurtosis of S_n, and $0 < \varepsilon < 1$, we stop resampling procedure.

3.2 Classification Results

We can now see that the C4.5 algorithm improves its classification performance when we apply it to rebalanced data sets. In our example, we have chosen $\gamma = \alpha = 5\%$. The percentage of correctly classified instances after resampling procedure have increased from 18% to 35% for Region 1, from 26% to 37% - for Region 2 (see below). Our approach also yields promising results in terms of the alternative performance measures.

```
CLASSIFICATION RESULTS FOR REGION 1:
Correctly Classified Instances 35.0365 %,
Recall = 0.3089, Precision = 0.5783,
G-mean = 0.3734, F-measure = 0.4027.

CLASSIFICATION RESULTS FOR REGION 2:
Correctly Classified Instances 37.5 %,
Recall = 0.3477, Precision = 0.6476,
G-mean = 0.3972, F-measure = 0.4525.
```

The histograms indicate approximately uniform distributions after procedure completion, with kurtosis magnitudes equal to 1.8643 and 1.7894, respectively.

Reporting our experimental results in dynamics, one can see in Figures 3-5, that the measures those we are interested in, are increasing (although non-monotonically) with the iteration number. In each figure, we plot every classification measure for both regions: percentage of correctly classified examples (Figure 3), Recall (Figure 4), Precision (Figure 5) (G-mean and F-measure plots are not presented here by lack of space).

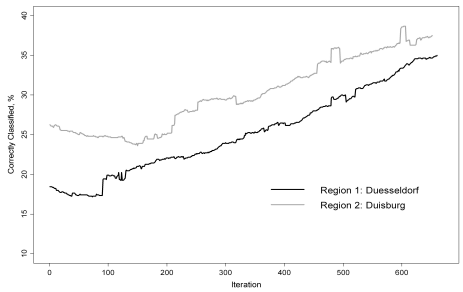


Figure 3: Algorithm Performance for Regions: Accuracy, %.

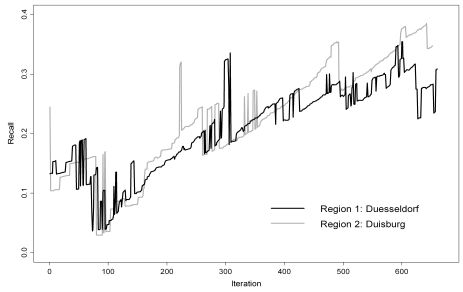


Figure 4: Algorithm Performance for Regions: Recall.

Note, that Precision lines are lying almost everywhere above all other lines. This tells us that proportion of

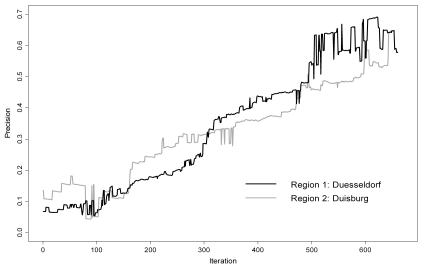


Figure 5: Algorithm Performance for Regions: Precision.

examples that were classified as elements from positive classes and those that are actually positive, grows most rapidly.

4 CONCLUSIONS AND FUTURE RESEARCH WORK

In this paper, a resampling technique based on statistical properties of data set, was proposed. We have tested our technique in terms of its accuracy and four performance measures: Recall, Precision, G-mean and F-measure. As investigation reveals, C4.5 algorithm applied to resampled data sets produced better results. But, in spite of the presented promising direction of rather general resampling techniques, the algorithm has to be yet improved in terms of classification performance. The effect of its application to various forms of data sets structure (highly skewed data sets, multimodal data sets, etc.) should be investigated as well. The comparison with other resampling methods also has to be carried out.

The resampling algorithm can be also carried out on the basis of the likelihood ratio test. The Neyman-Pearson Lemma implies that likelihood ratio test gives the best result in fixed size samples.

Further, for the start-up problem we were interested in, an accurate classification can result in injuries control boundaries analogous to presented in (Bondarenko, 2006a), (Bondarenko, 2006b), (F. Pokropp and Sever, 2006). The trees obtained by classification, can be very large (a lot of nodes and leaves), and in this since they are less comprehensible for control boundaries illustration. But we can simplify the obtained classification results by transforming every decision tree into a set of "if-then" rules ("Traffic Injuries Rules"), which seem to be easier for understanding and interpreting. Using real traffic injuries data, it is possible to develop realistic model for daily injuries number prediction, depending on temporal factors (year, month, day type). Of course, this research direction is open for other practical implications as well.

REFERENCES

Bondarenko, J. (2006a). Analysis of traffic injuries among children based on generalized linear model with a latent process in the mean. *Discussion Paper in Statistics and Quantitative Economics, Helmut-Schmidt University Hamburg*, (116).

Bondarenko, J. (2006b). Children traffic accidents models: Analysis and comparison. *Discussion Paper in Statistics and Quantitative Economics, Helmut-Schmidt University Hamburg*, (117).

F. Pokropp, W. Seidel, A. B. M. H. and Sever, K. (2006). Control charts for the number of children injured in traffic accidents. In H.-J. Lenz, P.-T. W., editor, *Frontiers in Statistical Quality Control*, pages 151–171. Physica, Heidelberg, 5 edition.

G. Cohen, M. Hilario, H. S. S. H. and Geissbuhler, A. (2005). Learning from imbalanced data in surveillance of nosocomial infection. *Artificial Intelligence in Medicine*, 37:7–18.

H. Han, W. W. and Mao, B. (2005). Borderline-smote: A new over-sampling method in imbalanced data sets learning. In *Proceedings of the International Conference on Intelligent Computing*, pages 878–887.

Hido, S. and Kashima, H. (2008). Roughly balanced bagging for imbalanced data. In *Proceedings of the SIAM International Conference on Data Mining*, pages 143–152.

Kohavi, R. and Provost, F. (1998). Glossary of terms. editorial for the special issue on applications of machine learning and the knowledge discovery process. *Machine Learning*, 30:271–274.

Kubat, M. and Matwin, S. (1997). Adressing the curse of imbalanced training sets: Onesided selection. In *Proceedings of the 14th International Conference on Machine Learning*, pages 179–186.

N. Chawla, K. W. Bowyer, L. O. H. and Kegelmeyer, W. P. (2002). Smote: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Quinlan, J. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, California.

S. Ertekin, J. H. and Giles, C. L. (2007). Active learning for class imbalance problem. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 823–824.

S. Kotsiantis, D. K. and Pintelas, P. (2006). Handling imbalanced datasets: a review. *GESTS International Transactions on Computer Science and Engineering*, 30:25–36.

V. Garcha, J. S. and Mollineda, R. (2008). On the use of surrounding neighbors for synthetic over-sampling of the minority class. In *Proceedings of the 8th WSEAS International Conference on Simulation, Modelling and Optimization*, pages 389–394.

X.-Y. Liu, J. W. and Zhou, Z.-H. (2006). Exploratory undersampling for class-imbalance learning. In *Proceedings of the International Conference on Data Mining*, pages 965–969.

MULTI-CORE COMPUTING UNIT FOR ARTIFICIAL NEURAL NETWORKS IN FPGA CHIP

Marek Bohrn and Lukas Fujcik
Department of Microelectronics, Brno University of Technology, Udolni 53, Brno, Czech Republic
bohrn@phd.feec.vutbr.cz, fujcik@feec.vutbr.cz

Keywords: Artificial neural networks, Computation acceleration, FPGA, VHDL, Spartan-3.

Abstract: This article describes a design and features of a multi-core unit for performing computing operations required for artificial neural network functioning. Its purpose is to speed up computing operations of the neural network. The number of computing cores can be altered as needed to achieve the required performance. VHDL language has been used to build this module. It has been optimized for the Spartan-3 family FPGA chips from Xilinx. These chips are favorable because of their low price and a high number of on-chip multipliers and block memory units. Spartan-3 chips facilitate parallel computing operations within neural networks to a very high level and thus help to achieve high computing power.

1 INTRODUCTION

Artificial neural networks are suitable for various tasks, such as pattern recognition, signal processing, classification, function approximation, prediction, data compression, etc.

It is possible to apply analog circuits, micro-controllers, computers, signal processors, and FPGA chips of neural networks implementation. Analogue neural networks are not used because of their implementation complexity and computing accuracy. Micro-controllers are cheap and easy to use, however, they do not achieve the required computing performance. Implementation of a neural network in a computer is easy, but it is unsuitable for use in industry or as embedded application due to the high cost and/or enormous size.

Signal processors and FPGA chips are suitable platforms for the implementation of neural networks. Signal processors are easier to use, they can be programmed in C language and they are very cheap for this purpose. Their disadvantage is that their computing power cannot be increased easily due to their fixed structure.

FPGA chips, on the other hand, have the capacity to facilitate parallel computing operations at a very high level and their structure can be adjusted to target application. The programming language used for FPGA chips is usually VHDL which is suitable for the description of parallel structures and pipe-line circuits. Application of these techniques allows for achieving the maximum computing power and speed and thus competes with all remaining alternatives of neural networks implementation.

The unit described in this article takes advantage of all benefits of FPGA chips and has been optimized for Spartan-3 family from Xilinx. The solution provided herein performs parallel computing operations and use pipe-line structure. The number of computational cores can be adjusted as needed. The biggest chip of Spartan-3 family (XC3S-5000) can employ up to 100 computational cores (Xilinx, 2006). The XC3S-200 chip was chosen to carry out the implementation and testing of the unit.

2 THEORETICAL ANALYSIS

2.1 Artificial Neuron

Artificial neuron – perceptron (Figure 1) is composed of an input vector, synaptic weight vector, summation unit, activation function and output.

Its function is as follows:

The input vector is multiplied by the weight vector. The results are summed up and they compose the inner potential of the neuron as depicted in Equation 1 (Fausett, 1994).

$$i = \sum_{n=1}^{x} w_n * in_n \qquad (1)$$

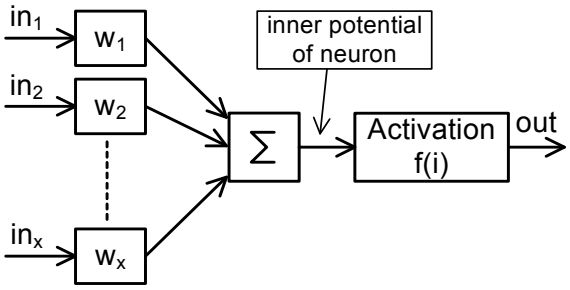


Figure 1: Structure of artificial neuron.

This value is fed into the activation function. The activation function is usually a step function, a sigmoid or ramp function; its task is to saturate the output value within the set limits, usually between 0 and 1.

2.2 Computations Inside Neural Network

Multiplication and summation are the most important computations within the neural network. These can be carried out using the multiply-accumulate (MAC) function, however, the chosen target chip Spartan-3 does not support this particular function. Instead, the MAC operation is replaced by sequential multiplication and summation operations. Spartan-3 chips include hardware multipliers and the summation operations are performed by gates.

The computation of inner potential of a neuron is always performed by one computational block. These blocks are in a parallel arrangement which increases the data throughput capacity as well as the overall computing power of the entire unit. The Spartan-3 chips can work up to 375 MHz under ideal conditions; however, the actual frequency of operation is slightly lower.

For the activation function, the unipolar sigmoid has been chosen (Figure 2) because it is the most common function for neural computations. The activation function is carried out using interpolation from the lookup table. It is possible to add other activation functions to the chip or replace the unipolar sigmoid by another function, if needed.

2.3 Format of Data

The data format for the weight vectors and input information transfer and the computational operations are the same within the neural network. The 18-bit mode is the most suitable width for buses employed in the Spartan-3 chips because of the 18-bit inputs of the embedded multipliers. The block

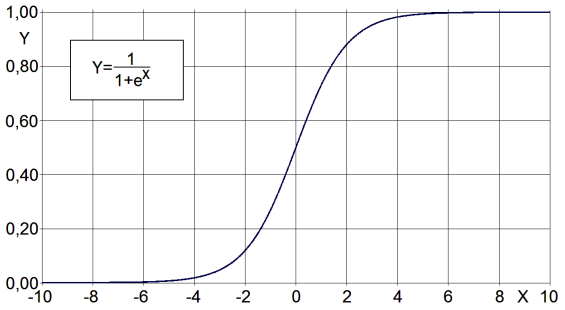


Figure 2: Sigmoid activation function.

memory units can also be adjusted to the 18-bit mode.

The data are represented as a fixed point number and are stored in two's complement. The 6 most significant bits of the data word represent the integer part and the next 12 bits represent the fraction part as illustrated in Figure 3 (Suhap, Becerikili, Yazici, 2006).

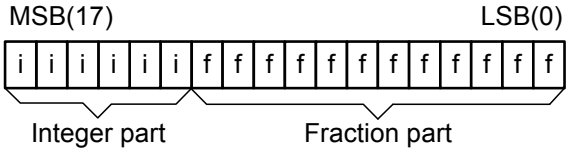


Figure 3: Format of number representation.

3 REALIZATION IN FPGA CHIP

3.1 Circuit Overview

Figure 4 shows a block diagram of the entire circuit for neural network computations. The control logic block, neuron computation blocks, and activation function are the main parts of the circuit.

The control logic includes a map of the neural network which is the basis of data and commands sent out to neuron computation blocks. The selection of weight is done by an address sent to the bus controlling the memory blocks. Weight coefficients are stored in these memory blocks. Each memory block can store up to 1,024 weight coefficients. In the XC3S-200 chip used for testing the design, up to 10,240 synapses can be implemented.

Data coefficients are transmitted on the basis of an external requirement and a neural network computational status. Either data stored in the neuron data memory or data from the input port are used. The selected neuron data memory configuration allows for storage of 1,024 neuron

outputs which means that the unit can handle a neural network consisting of up to 1,024 neurons.

The control logic can also send the following commands to the computational blocks: reset, MAC, BIAS, send result. Each neuron computation block is connected to an individual command bus. By separating the command buses of individual neurons, it is possible to arrange them in a way so that each neuron performs computational operations simultaneously at a different layer of the neural network. Possibly some of the neuron cores can be disconnected by sending only the reset command. MAC command together with BIAS command calculates the inner potential of the given neuron. Once the commands are received by the neuron cores, the computational core starts to multiply the weight coefficients by data inputs and stores the results in the internal register. The send result command initiates the transfer of the inner potential value to the next register. Subsequently, the neuron waits for an impulse to send the data to the activation function.

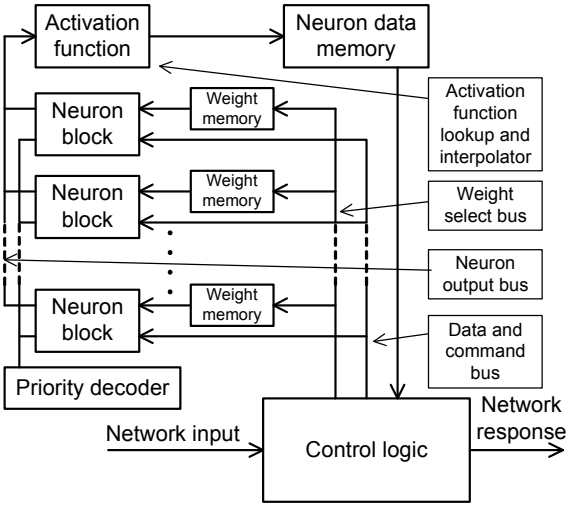


Figure 4: Block diagram of a complete unit.

The output bus of neuron computing blocks is shared by all computing blocks because its capacity utilization should be the low. When required to send the data to the activation function, neurons wait for the priority decoder's impulse. Data are sequentially brought from computing blocks to the activation function and no internal FIFO buffer implementation in necessary.

After the data pass through the activation function, they are stored in the neuron data memory - dual port RAM. If needed they are read and sent from this memory via the data bus to the computing blocks. If the stored data belong to the last neuron layer, they are transferred to the unit output.

3.2 Neuron Computation Block

The block diagram of a neuron computing block is illustrated in the Figure 5. The block is arranged as a three stage pipe-line circuit to increase the speed. The input data of this block are command, data and weight.

In the first stage, the input data are multiplied according to the chosen command, the second stage is composed as a summation block. These two together implement the MAC function and calculate the internal neuron potential. When the send result command is received, the internal potential is sent to the last pipe-line stage and waits for the impulse to be transferred to the activation function.

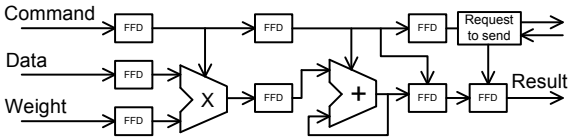


Figure 5: Block diagram of a neuron computation block.

After receiving the reset command, the internal potential is set to zero and the block can start computing another neuron regardless of the fact whether the result was sent or is waiting in the last stage, the result remains in an unchanged state.

3.3 Activation Function

Activation function is composed by an interpolated lookup table. The activation function circuit structure is pipe-line similar to computing block, it includes a multiplier and an adder.

The activation function has only one input which is divided into two parts. The upper part (bits 17 .. 9) is transferred into two lookup memory units. The lookup value for interval offset is stored in one of them and gradient of interval in the other. Gradient is multiplied by the lower part of the input (bits 8 .. 0) and is transferred to the summation part. Offset is delayed by a one clock period and it is also transferred to the summation block. The interpolated value of the activation function connected to the output is achieved in this way. The activation function output is stored in the neuron data memory which can be accessed by the control logic.

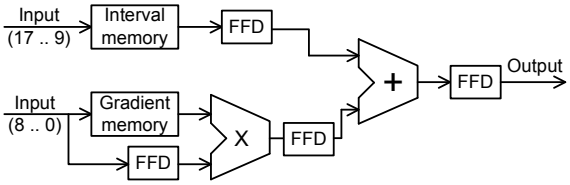


Figure 6: Block diagram of an activation function interpolator.

Since the activation function is realized by a lookup table, it is easy to change its shape using different coefficients. The circuit function can be extended by adding another block with a different activation function. For example, the activation function for output layer will be a step function if we require binary result.

3.4 Testing of Complete Circuit

The circuit was described in VHDL language, the source code is modular and optimized for the Spartan-3 family chips from Xilinx and fully synthesizable. Since most of the circuit is arranged into pipe-lines, it can work with the clock frequency up to 133MHz.

The target XC3S-200 chip includes 12 blocks of RAM and 12 dedicated multipliers. Activation function requires at least one block RAM and one multiplier. Neuron data memory needs one block RAM. For this reason it is possible to implement 10 computing blocks, needing one block of on-chip RAM for storing weight coefficients and one multiplier for computing.

The circuit function was tested on an application recognizing hand-written numbers. A network with 88 neurons in the input layer, 40 neurons in the hidden layer and 10 in the output layer was realized. The network model and the training algorithm were realized in a computer and weight coefficients were transferred to weight memory blocks on FPGA chip (Masters, 1993).

Calculation of response of the neural network require 88 times 4 clock periods for the hidden layer, 40 clock periods for the output layer plus latency of the activation function of 4 periods. For one character recognition, 396 clock cycles were needed. At the working frequency of 133 MHz the circuit can recognize 336000 numbers per a second. In comparison to the single core signal processor which would need for such a calculation at least 3920 clock cycles, this is an excellent result.

4 CONCLUSIONS

The neural network function is very demanding regarding to the computing power. In the FPGA chips, however, it is possible to parallelize the calculations very efficiently. The designed circuit speeds up the computation for neural network approximately 10 times in comparison to signal processor. With using the biggest chip XC3S-5000 from the Spartan-3 family, it would be possible to implement up to 100 computing cores into the circuit and thus increase the computing power theoretically up to 100 times.

The entire circuit is very modular and allows for realization of neural networks in various configurations. From very small networks up to networks consisting of thousands of neurons and hundreds of thousands synapses with a very high computing power kept.

Spartan-3 family chip was chosen for the implementation because of its low price and good accessibility of development kits. Its disadvantage is that it cannot process the MAC command in one clock cycle. This problem is solved by higher FPGA families, such as Virtex-5 family chips are able to process up to 580 Giga MAC per second while their computing of MAC command is performed within one clock cycle.

ACKNOWLEDGEMENTS

This research has been supported by the Czech Ministry of Education in the frame of long term Program Plan MSM 0021630503 MIKROSYN New Trends in Microelectronic Systems and Nanotechnology and by the Czech Science Foundation as the project GA102/08/1116.

REFERENCES

Ohomondi, Amos, R., Rajapasake, Jagath, C., 2006. *FPGA Implementations of Neural Networks*, Springer Netherlands

Fausett, L., 1994. *Fundamentals of Neural Networks*, Prentice Hall New Jersey

Masters, T., 1993. *Practical Neural Network Recipes in C++*, Academic Press California

Xilinx, 2006. Spartan-3 FPGA Family: Complete Data Sheet, Xilinx company

Suhap, S., Becerikili, Y., Yazici, S., 2006. *Neural Network Implementation in Hardware Using PFGAs, 13th International Conference, ICONIP 2006*, Springer-Verlag Germany

A MULTI-OBJECTIVE APPROACH TO APPROXIMATE THE STABILIZING REGION FOR LINEAR CONTROL SYSTEMS

Gustavo Sánchez, Miguel Strefezza
Universidad Simón Bolivar, Departamento de Procesos y Sistemas, Venezuela
gsanchez,strefeza@usb.ve

Orlando Reyes
Universidad Simón Bolivar, Departamento de Tecnologia Industrial, Venezuela
orlandoreyes@.usb.ve

Keywords: Stabilizing Region, Controller Design, PID Control, Randomized Algorithms, Genetic Algorithms.

Abstract: Stability is a crucial issue to consider for control system design. In this paper a new multi-objective approach to approximate the stabilizing region for linear control systems is proposed. The design method comprises two stages. In the first stage, a bi-objective sub-problem is solved: the algorithm aims to calculate the vertices that maximize both the volume of the decision space and the percent of stable individuals generated within the decision space. In the second stage, the information gathered during the first stage is used to solve the actual multi-objective control design problem. To evaluate the proposed method a PID design problem is considered. Results show that in this case, our method is able to find better Pareto approximations than the classical approach.

1 INTRODUCTION

Multi-Objective Evolutionary Algorithms (MOEA) have been succesfully applied to control applications, provided the optimization problem can not be solved by classical methods (Fleming, 2001),(Tan et al., 2005). However, these algorithms often operate in a fixed search space predefined *a priori*, which hopefully contains the stabilizing region (Khor et al., 2002). Note that this requirement is hard to fulfil in practical control applications, since in general there is little information about the geometry of the stabilizing region. However, the stability of a control system is extremely important and is generally a safety issue.

Evolutionary computation techniques assume the existence of an efficient evaluation of the feasible individuals. However, there is no uniform methodology for evaluating unfeasible ones. In fact, several techniques have been proposed to handle constraints within the framework of MOEAs. The simplest approach is for example to reject unfeasible solutions (Michalewicz, 1995). In this work, we propose to determine an approximation of the stabilizing region trying to generate and to keep feasible individuals during the whole search process. Note that in practice the information about the stabilizing region is very important for the control engineer.

Quite surprisingly, there exist few references about MOEAs which consider a dynamic search space. As an example, a multi-population algorithm called Multi-Objective Robust Control Design (MRCD) was proposed in (Herreros, 2000). The search space is dynamic and a filtered elite is employed to preserve the best results for future generations.

In (Arakawa et al., 1998) an algorithm called Adaptive Range Genetic Algorithm (ARange GA) was described. The searching range is adapted in such a way that there is no need to define the initial range and nor the number of bits to determine the precision of results, if binary codification is adopted.

In (Khor et al., 2002) an inductive/deductive learning approach for single and multi-objective evolutionary optimization was proposed. The method is able of directing evolution towards more promising search regions even if these regions are outside the initial predefined space. For problems where the global optimum is included in the initial search space, it is able of shrinking the search space dynamically for better resolution in genetic representation.

Another algorithm called ARMOGA (Adaptive Range Multi-Objective Genetic Algorithm) was pro-

posed in (Sasaki et al., 2002). Based on the statistics of designs computed so far, the new decision space is determined. The authors claim that their algorithm makes possible to reduce the number of evaluations to obtain Pareto solutions.

In other hand, randomized techniques for analysis of uncertain systems have attracted considerable interest in recent years, and a significant amount of results have appeared in the literature (Calafiore and Dabbene, 2006). The basic idea is to characterize uncertain parameters as random variables, and then to evaluate the performance in terms of probabilities. Analogously, probabilistic synthesis is aimed at determining the parameters so that certain desired levels of performance are attained with high probability.

In the following, a new randomized multi-objective approach to determine an approximation of the stabilizing region for linear controllers is presented. The design method comprises two stages. In the first stage a bi-objective sub-problem is solved: the vertices that maximize both the volume of the decision space and the ratio of feasible individuals are calculated. In the second stage, the information gathered during the first stage is used to solve the actual multi-objective control design problem.

To evaluate the proposed method a **PID** design problem is considered. Note that these controllers are still the preferred structure in most industries. And this is because of several reasons. From a practical view, they come pre-programmed in most commercial controllers and are well understood by both engineers and technicians. Thus, it is important to verify whether any design problem can be solved using a **PID** structure, before testing more complex solutions.

In the framework of this paper, we have another important motivation to consider **PID** controllers. Recently, Hohenbichler *et al* (2008) developed a **MATLAB**® toolbox, called **PIDROBUST**, which allows to calculate the stability region of control loops which contain an ideal **PID** controller. This is why we propose to evaluate the quality of the the boundaries which are going to be estimated in this paper by means of the new proposed randomized multi-objective approach.

The paper is organized as follows. In section 2, the design problem is formulated. In section 3 the proposed design method is described. In section 4, numerical results are given and finally in section 5 conclusions are given.

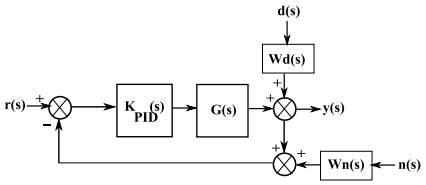


Figure 1: Control loop with a PID controller.

2 PROBLEM FORMULATION

Consider the control loop in figure 1. Therein, the ideal **PID** controller has the equation shown in (1). Blocks $W_n(s)$ and $W_d(s)$, $s \in \mathbb{C}$ represent arbitrary filters. The signals $r(s), d(s)$ and $n(s)$ represent the reference, disturbance and noise signals respectively. For regulation problems it is assumed $r(s) = 0$ and then equations (2), (3) and (4) hold.

$$K_{PID}(s) = \frac{K_D s^2 + K_P s + K_I}{s} \tag{1}$$

$$y(s) = -T(s)W_n(s)n(s) + S(s)W_d(s)d(s) \tag{2}$$

$$T(s) = \frac{K_{PID}(s)G(s)}{1 + K_{PID}(s)G(s)} \tag{3}$$

$$S(s) = \frac{1}{1 + K_{PID}(s)G(s)} \tag{4}$$

The transfer functions $S(s)$ and $T(s)$ are known as sensitivity function and complementary sensitivity function respectively. To attenuate disturbance and noise signals, both $S(s)$ and $T(s)$ must be made "small". However, note that from equations (3) and (4) it is impossible to decrease the norms of both $S(s)$ and $T(s)$ simultaneously: this is why we need to seek the best trade-off solutions. For simplicity, in this work only two $\mathcal{H}_2$ objectives are considered and the control problem can be stated as follows:

$$\min_{K_P, K_D, K_I \in \mathbb{R}} \begin{pmatrix} \|W_d S(K_P, K_D, K_I)\|_2 \\ \|W_n T(K_P, K_D, K_I)\|_2 \end{pmatrix} \tag{5}$$

subject to

$$K_{PID} = \frac{K_D s^2 + K_P s + K_I}{s} \in \mathcal{K}$$

where $\mathcal{K}$ is the set of stabilizing controllers.

3 DESIGN METHOD

To solve (5) a common approach is to define a penalty function to avoid unstable individuals. For example,

a penalty function $p_i(K_{PID})$ $i \in \{1,2\}$ can be defined for each objective function as:

$$p_i(K_{PID}) = \begin{cases} f_i(K_{PID}) \text{ if } \max\{\mathrm{Re}[\lambda(K_{PID})]\} < -\mu \\ \eta * \max\{\mathrm{Re}[\lambda(K_{PID})]\} + B, \text{otherwise} \end{cases} \quad (6)$$

where $\lambda(K_{PID})$ are the poles of $S(K_{PID})$, μ, η, B are fixed parameters and

$$f_1(K_{PID}) = \|W_d S(K_P, K_D, K_I)\|_2 \quad (7)$$

$$f_2(K_{PID}) = \|W_n T(K_P, K_D, K_I)\|_2 \quad (8)$$

Although simple, the penalty approach can be inefficient and lead to premature convergence. We propose instead to estimate an approximation of $\mathcal{K}$ (see the flowchart shown in figure 2). The proposed design method comprises two stages. During the first stage the goal is to calculate the vertices that maximize both the volume of the decision space and the percent of stable individuals. Note that these two objectives are in conflict.

Let n be the number of decision variables for problem (5) and $\overline{L}, \underline{L} \in \mathbb{R}^n$ be vectors defining the range for each variable. As an example, for the **PID** problem, we have $n = 3$ and

$$\begin{aligned} K_P &\in [\underline{L}_1, \overline{L}_1] \\ K_I &\in [\underline{L}_2, \overline{L}_2] \\ K_D &\in [\underline{L}_3, \overline{L}_3] \end{aligned} \quad (9)$$

The following sub-problem is considered during the first stage:

$$\min_{\overline{L},\underline{L} \in \mathbb{R}^n} \begin{pmatrix} g_1(\overline{L}, \underline{L}) \\ g_2(\overline{L}, \underline{L}) \end{pmatrix} \quad (10)$$

where

$$g_1(\overline{L}, \underline{L}) = 1 - \frac{N_s(\overline{L}, \underline{L})}{N(\overline{L}, \underline{L})} \quad (11)$$

$$g_2(\overline{L}, \underline{L}) = \frac{V_{\max} - \prod_{i=1}^{n} (\overline{L}_i - \underline{L}_i)}{V_{\max}} \quad (12)$$

$$V_{\max} = \max_{\overline{L},\underline{L} \in \mathbb{R}^n} \prod_{i=1}^{n} (\overline{L}_i - \underline{L}_i)$$

and

$$N_s(\overline{L}, \underline{L}) = \sum_{i=1}^{N(\overline{L},\underline{L})} S_t(K_{PID_i}) \quad (13)$$

$$S_t(K_{PID}) = \begin{cases} 0 \text{ if } \max\{\mathrm{Re}[\lambda(K_{PID})]\} \geq -\mu \\ 1 \text{ otherwise} \end{cases} \quad (14)$$

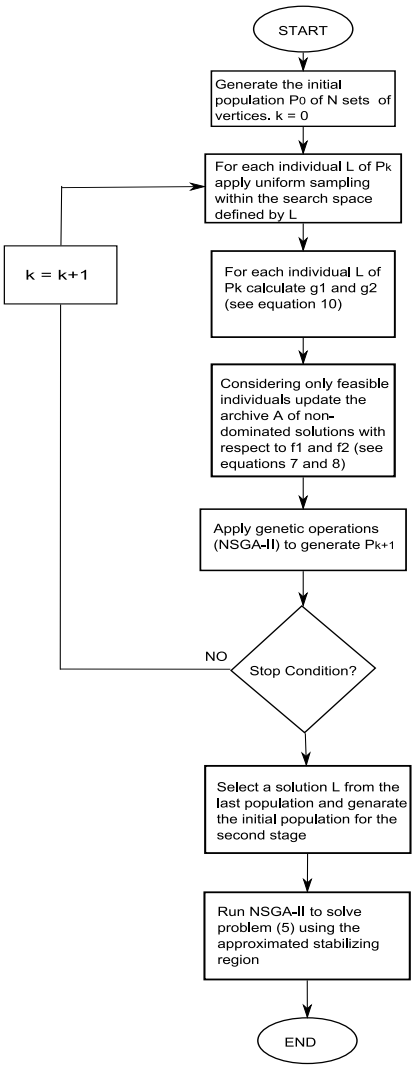


Figure 2: Flowchart of the proposed design method.

In (11) N_s is the number of stable individuals obtained via uniform sampling with boundaries $\overline{L}, \underline{L}$. The sample size $N(\overline{L}, \underline{L})$ may be calculated using the following Chernoff bound:

$$N(\overline{L}, \underline{L}) = \frac{1}{2\varepsilon^2} \ln \frac{2}{\delta} \quad (15)$$

It can be shown that if $N(\overline{L}, \underline{L})$ is taken according to (15) and the sampling is uniform the following equation holds:

$$\mathrm{Prob}\left\{ \left| p - \frac{N_s}{N} \right| \leq \varepsilon \right\} \geq 1 - \delta \quad (16)$$

where

$$p = \mathrm{Prob}\{S_t(K_{PID}) = 1\} \quad (17)$$

We have fixed ε and δ to 0.1. Note that during the first stage an archive of non-dominated solutions

can be updated, to be used later. In both stages any available MOEA can be used to solve the optimization problem. In this work we have chosen **NSGA-II** (Deb et al., 2000), which is a "state of the art" multiobjective genetic algorithm. **NSGA-II** is an improved version of the original algorithm proposed in (Srinivas and Deb, 1994). Note that this algorithm was previously used to design **PID** structures in (Lagunas, 2004). It has the following features:

- It uses an elitist principle.
- It uses and explicit diversity preserving mechanism.
- There is no sharing parameter to select.
- The sorting mechanism is faster than MOGA.

The offspring is created using the parent population and usual genetic operators. Thereafter, the two populations are combined together and a non-dominated sorting mechanism is used to classify the entire population.

Special mutation and cross-over operators were designed to generate individuals that remain within the limits of the (approximated) feasible region.

4 NUMERICAL RESULTS

Consider the open-loop model taken from (Hohenbichler and Abel, 2008):

$$G_0(s) = \frac{-0.5s^4 - 7s^3 - 2s + 1}{s^6 + 11s^5 + 46s^4 + 95s^3 + 109s^2 + 74s + 24} \quad (18)$$

and the filters W_n and W_d are defined:

$$W_n(s) = \frac{1}{s+1}, W_d(s) = 1 \quad (19)$$

For this plant, figure 3 shows an example of the convex stability polygons corresponding to $K_P = 1$. **PIDROBUST** allows also to plot the whole stabilizing region, as shown in figure 4.

The proposed design method was coded in **MATLAB**® 7.0. Figure 5 shows a typical Pareto approximation (Coello et al., 2007) obtained at the end of stage 1. Note that the Pareto front approximation shows that there exist no compromise solution between the two objectives. In this work the following solution have been arbitrary chosen:

$$\begin{aligned} [\underline{L}_1, \overline{L}_1] &= [-18.3, 6.1] \\ [\underline{L}_2, \overline{L}_2] &= [-4.3, 9.69] \\ [\underline{L}_3, \overline{L}_3] &= [-83.3, 14.3] \end{aligned} \quad (20)$$

which corresponds to $g_1 \approx 0.2$ and $g_2 \approx 0.99$.

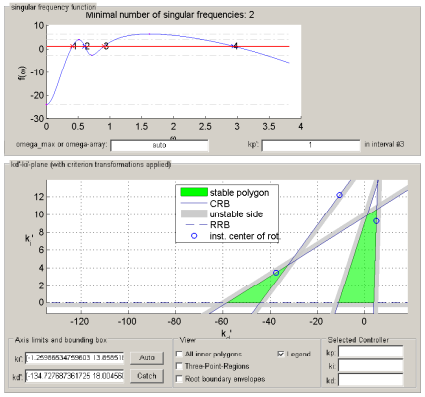


Figure 3: Example of the convex stability polygons corresponding to $K_P = 1$.

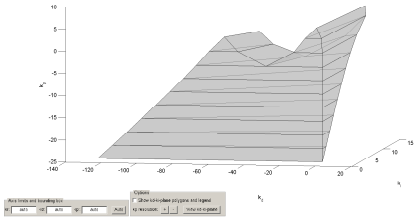


Figure 4: Stabilizing region.

To compare the results, the design problem was solved by means of the classical "penalty function based approach (in this case **NSGA-II** as a standalone algorithm), with $\mu = 10^{-4}, \eta = 10^3, B = 10^3$ and generating the individuals within the cube defined by:

$$K_P, K_D, K_I \in [-200, 200] \quad (21)$$

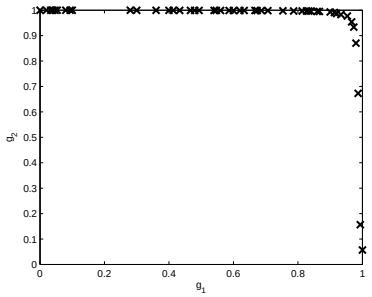


Figure 5: Pareto approximation obtained at the end of the first stage.

The comparison is fair because this region envelopes the true stabilizing region, although this information is not available *a priori*. We also include the results corresponding to the true stabilizing region

obtained by **PIDROBUST**, with boundaries:

$$\begin{aligned}[\underline{L}_1,\overline{L}_1] &= [-22.5,4.9] \\ [\underline{L}_2,\overline{L}_2] &= [0,12.59] \\ [\underline{L}_3,\overline{L}_3] &= [-118,5.24]\end{aligned} \tag{22}$$

Thus, three algorithms are compared:

- **A1** : NSGA-II with $K_P, K_D, K_I \in [-200,200]$.
- **A2** : NSGA-II: Stage 1 and Stage 2 (proposed new method) with $\varepsilon = 0.1$ and $\delta = 0.1$
- **A3** : NSGA-II + PIDROBUST (using the true stabilizing region).

Figures 6, 7 and 8 show typical approximations obtained via **A1**, **A2** and **A3** respectively. In the multi-objective optimization framework, there are in general three goals (i) the distance of the resulting nondominated set to the Pareto-optimal front should be minimized, (ii) a uniform distribution of the solutions found is desirable, and (iii) the extent of the obtained nondominated front should be maximized. We have chosen four indicators to evaluate each Pareto approximation:

- Set Coverage (C). Given two Pareto approximations $\mathcal{PF}_1$ and $\mathcal{PF}_2$, the Set Coverage is defined as follows:

$$C(\mathcal{PF}_1,\mathcal{PF}_2) = \frac{|\{b \in \mathcal{PF}_2, \exists a \in \mathcal{PF}_1 : a \preceq b\}|}{|\mathcal{PF}_2|} \tag{23}$$

  $C(\mathcal{PF}_1,\mathcal{PF}_2)$ measures the proportion of elements in $\mathcal{PF}_2$ that are dominated by at least one element in $\mathcal{PF}_1$.

- Efficient Set Space (ESS). Defined as follows:

$$ESS = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(d_i-\overline{d})^2} \tag{24}$$

  Hereby, d_i denotes the minimal Euclidean distance from the image $\mathcal{PF}_1(x_i)$ of a solution $x_i, i = 1,\ldots,N$, to the true Pareto front, and

$$d_i := \min_{\substack{j=1,\ldots,N \\ i\neq j}} d_{ij} \tag{25}$$

$$\overline{d} := \frac{1}{N}\sum_{i=1}^{N} d_i, \tag{26}$$

  where d_{ij} is the Euclidean distance between $\mathcal{PF}_1(x_i)$ and $\mathcal{PF}_1(x_j)$

- Hypervolume (HV). This indicator measures the volume covered by the approximation with respect to a reference point.
- Max. Distance (MD). Defined as:

$$MD = \max_{\substack{i,j=1,\ldots,N \\ i\neq j}} d_{ij} \tag{27}$$

The setting used for each algorithm is shown in table 1. Tables 2 and 3 show the indicators values for 30 executions. The mean value and the standard deviation (between brackets) is presented for each indicator. Table 4 shows the results of Mann-Whitney-Wilcoxon tests with a significance level $\alpha = 0.05$. Therein, the symbols $=, \Uparrow, \Downarrow$ mean that the algorithm in the corresponding row is statistically equal, better or worse than the algorithm in the column, with respect to the four indicators C, ESS, HV and MD.

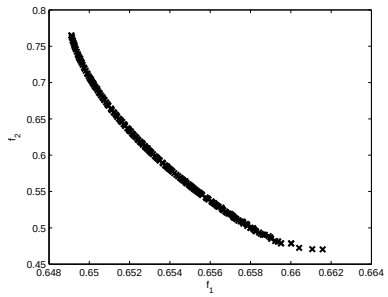


Figure 6: Example of Pareto approximation obtained with algorithm A1.

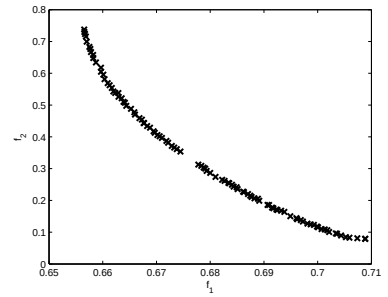


Figure 7: Example of Pareto approximation obtained with algorithm A2.

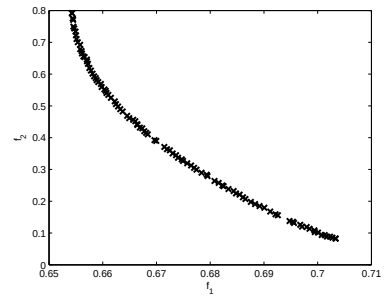


Figure 8: Example of Pareto approximation obtained with algorithm A3.

Table 1: Parameter Settings NSGA-II.

Representation	Real numbers
Cross-Over Operator	Uniform
Cross-Over Probability	0.9
Mutation	Gaussian Perturbation
Mutation Probability	0.1
Stop Condition A1	300 generations
Selection Scheme A1	(200+200)
Stop Condition A2,A3	50 generations
Selection Scheme A2,A3	(100+100)

Table 2: Results for the coverage indicator.

$C(A_i,A_j)$	**A1**	**A2**	**A3**
A1	–	0.26(0.31)	0.23(0.33)
A2	0.26(0.43)	–	0.20(0.33)
A3	0.23(0.39)	0.56(0.35)	–

Table 3: Results for indicators ESS, HV and DMAX.

	ESS	*HV*	*MD*
A1	0.0452(0.05)	0.03(0.03)	0.03(0.03)
A2	0.14(0.08)	0.13(0.00)	0.15(0.09)
A3	0.11(0.07)	0.10(0.02)	0.11(0.07)

Table 4: Results for Mann-Whitney-Wilcoxon tests with a significance level $\alpha = 0.05$.

	A1	**A2**	**A3**
A1	–	$=,\Uparrow,\Downarrow,\Downarrow$	$=,\Uparrow,\Downarrow,\Downarrow$
A2	$=,\Downarrow,\Uparrow,\Uparrow$	–	$\Downarrow,\Downarrow,\Uparrow,\Downarrow$
A3	$=,\Downarrow,\Uparrow,\Uparrow$	$\Uparrow,\Uparrow,\Downarrow,\Uparrow$	–

5 CONCLUSIONS

In this paper a new design method for solving multi-objective control problems was described. Results show that, for the PID control problem, the proposed method **(A2)** is able to find better approximations than the conventional method (**A1**) with respect to the *HV* and *MD* indicators, given the same number of functions evaluations. Moreover, when compared to **A3** (the algorithm using the true stabilizing region), results show that **A2** is indeed able to find a good approximation of the feasible region.

In the future the authors will try to extend this work, in order to consider more complex control problems, with more design variables and objectives. For that goal, a more sophisticated representation of the search space is needed (stage 1), in order to make the sampling process more efficient.

REFERENCES

Arakawa, M., Nakayama, H., Hagiwara, I., and Yamakawa, H. (1998). Multiobjective Optimization Using Adaptive Range Genetic Algorithms with Data Envelopment Analysis. In *A Collection of Technical Papers of the 7th Symposium on Multidisciplinary Analysis and Optimization*, pages 2074–2082.

Calafiore, G. and Dabbene, F. (2006). *Probabilistic and Randomized Methods for Design under Uncertainty*. Springer-Verlag,London.

Coello, C., Lamont, G., and Van-Veldhuizen, D. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer.Genetic and Evolutionary Computation Series.Second Edition.

Deb, K., Agrawal, S., Pratab, A., and Meyarivan, T. (2000). A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pages 849–858.

Fleming, P. (2001). Genetic Algorithms in Control System Engineering. Technical report, Research Report No. 789. University of Sheffield.UK.

Herreros, A. (2000). Diseo de controladores robustos multiobjetivo por medio de algoritmos geneticos. *Tesis Doctoral, Departamento de Ingenieria de Sistemas y Automatica.Universidad de Valladolid, Espana.*

Hohenbichler, N. and Abel, D. (2008). Calculating All KP Admitting Stability of a PID Control Loop. In *Proceedings of the 17th World IFAC Congress. Seoul, Korea, July 6-11*, pages 5802–5807.

Khor, E., Tan, K., and Lee, T. (2002). Learning the Search Range for Evolutionary Optimization in Dynamic Environments. *Knowledge and Information Systems Vol. 4 (2)*, pages 228–255.

Lagunas, J. (2004). *Sintonizacin de controladores PID mediante un algoritmo gentico multi-objetivo (NSGA-II)*. PhD thesis, Departamento de Control Automtico. Centro de Investigacin y de Estudios Avanzados. Mxico.

Michalewicz, Z. (1995). A survey of constraint handling techniques in evolutionary computation methods. In *Proceedings of the 4th Annual Conference on Evolutionary Programming*, pages 135–155. MIT Press.

Sasaki, D., Obayashi, S., and Nakahashi, K. (2002). Navier-Stokes Optimization of Supersonics Wings with Four Objectives Using Evolutionary Algorithms. *Journal of Aircraft, Vol. 39, No. 4, 2002, pp. 621629*.

Srinivas, N. and Deb, K. (1994). Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation. Vol 2. No 3*, pages 221–248.

Tan, K., Khor, E., and Lee, T. (2005). *Multiobjective Evolutionary Algorithms and Applications*. Springer-Verlag.

A MIN-PLUS APPROACH FOR TRAFFIC FLOW MODELING

Julien Rousseau, Sébastien Lahaye
LISA, 62 avenue Notre Dame du Lac, 49000 Angers, France
julien.rousseau@etud.univ-angers.fr, lahaye@univ-angers.fr

Claude Martinez
IRCCYN, 1 rue de la Noë, 44000, Nantes, France
claude.martinez@irccyn.ec-nantes.fr

Jean-Louis Boimond
LISA, 62 avenue Notre Dame du Lac, 49000 Angers, France
boimond@univ-angers.fr

Keywords: Min-plus algebra, Min-plus linear systems, Vehicular traffic flow modeling, Traffic flow simulation.

Abstract: In this paper we propose a modeling method for traffic flow phenomena based on the min-plus algebra. We adopt a modular approach by dividing roadways as elementary stretches which can be combined in order to get a model for a complex infrastructure. The approach is flexible in the sense that different scales can be considered for each elementary model. In fact, whatever its size, each roadway stretch is here studied as a min-plus linear system and is modeled by its impulse response in min-plus algebra. In this first step in studying traffic flow, we focus on modeling detailing the adopted methodology. We also present simulations to validate the approach.

1 INTRODUCTION

Traffic is a non-linear phenomenon complex to predict or simulate. In spite of that, many models for traffic flow were studied since the late fifties, some at a macroscopic level (for example models based on gas kinetic, see (Helbing, 1996)), others at a microscopic level (for example models based on cellular automata, see (Nagel and Schreckenberg, 1992)).
In this paper we propose a modeling method for traffic flow phenomena based on the min-plus algebra. Previous works have used min-plus algebra to study road traffic (Lolito et al., 2005), (Farhi et al., 2007). In these papers, authors present a microscopic model based on Petri nets whose dynamics are written using min-plus algebra. This approach is not far from cellular automata approach (Nagel and Schreckenberg, 1992) since roadways are divided in stretches containing at most one vehicle, and individual movement of a vehicle in a stretch is conditioned by its availability (no vehicle on this section).
The proposed approach is more flexible in the sense that different scales can be considered for the model. We also adopt a modular approach by dividing roadways as elementary stretches which can be combined in order to get a model for a complex infrastructure. But the scale of a stretch can be larger. In fact, whatever its size, each roadway stretch is here studied as a min-plus linear system and is modeled by its impulse response in min-plus algebra. In the extreme, a stretch can be sized to contain a single vehicle. It can also correspond to a more macroscopic element (*e.g.* a roadway which is several kilometers long). Doing so, we expect that descriptions of various phenomena (inherent to various roadway configurations such as intersections, traffic lights,...) can be modulated according to their complexity and that large problems can be tackled. Whereas analytical results were derived in (Lolito et al., 2005), (Farhi et al., 2007), we here focus on modeling of traffic flow. In fact, this first paper aims only at stating the adopted modeling methodology. We present simulations to validate the approach.
The paper is organized as follows.
In section 2, we show that vehicular traffic flow can be considered as a min-plus linear system, that is a linear system over min-plus algebra. Representation of such a system thanks to its impulse response is recalled.

In section 3, we propose a methodology to model traffic flow as a min-plus linear system. A particular 2 inputs-2 outputs system is presented as a generic representation for a wide-variety of elementary roadway stretches. Then we show how such elementary models can be composed to obtain a model for a succession of roadway sections.
In section 4, two examples are presented. The first one concerns a basic road section (without any intersection). The proposed model is simple but rough in the sense that it leads to considered that an infinity of vehicles can simultaneously run on the section. The second example is a refinement (taking into account the limited capacity), and the composition of two elementary models is experimented. In both cases, we derive from simulation the *fundamental diagram* that links the flow to the density of vehicles on the road.
In section 5, we discuss characteristics of the proposed model comparatively with traffic flow models in the literature.

2 PRELIMINARIES

In this section, we first explain why vehicular traffic flow on a road section can be studied as a linear system over min-plus algebra. Then we recall that min-plus linear systems, and in particular traffic flow, can be represented by their impulse responses.

2.1 Traffic on a Roadway Stretch is a Min-plus Linear System

As usual when studying complex systems, we consider a roadway network as an assembly of road sections. Each roadway stretch will be seen as a min-plus linear system, and so, a complex infrastructure will be studied as the system resulting from the assembly of corresponding elementary subsystems. In other words, each road section is considered to be a simple min-plus linear system whose input and output correspond to the flows of vehicles respectively entering and leaving the section.

Let us recall that min-plus linear systems are systems for which the property of linearity, also called "principle of superposition", can be applied to the two binary operations min and $+$ of the min-plus algebra (see for example (Gaubert, 1992)).

Definition 2.1 (Signal, Min-plus Linear System) *A signal u is defined as a map from $\mathbb{Z}$ to $\mathbb{R}\cup\{-\infty\}$.*
A system $\mathcal{S}$ is called min-plus linear if for all signal u,v, and $\forall t\in\mathbb{Z}$,

$$[\mathcal{S}(\min(u,v))](t) \quad = \quad \min([\mathcal{S}(u)](t),[\mathcal{S}(v)](t)), \quad (1)$$

and $\forall u, \forall a\in\mathbb{R}, \forall t\in\mathbb{Z}$,

$$[\mathcal{S}(a+u)](t) \quad = \quad a+[\mathcal{S}(u)](t). \quad (2)$$

in which $\mathcal{S}(u)$ is the system output signal in response to input u.

To study a road section as a min-plus linear system, let us give the following meanings for its input and output :

- the input u is a counter of vehicles entering the road section: $u(t)$ denotes the cumulated number of vehicles having entered the road section up to time t,
- the output y is a counter of vehicles leaving the road section: $y(t)$ denotes the cumulated number of vehicles having left the road section up to time t.

It is assumed that vehicles are conserved along a road section[1], *i.e.*, $u(t)$ is always equal to the sum of $y(t)$ and the number of vehicles on the road section at time t.

Under this assumption, the amount of vehicles out of the road section with $\min(u,v)$ as input flow, is claimed to be equal to the minimum of quantity of vehicles out of the road section obtained with u and v considered separately.
On the one hand, since $min(u,v)$ corresponds to a less dense flow than the ones given separately by u and v, vehicles of the flow given by $min(u,v)$ cross the road section at least as fast as the ones in flows u and v considered separately. So we deduce the following inequality:

$$\forall t\,,\ [\mathcal{S}(min(u,v))](t)\geq min([\mathcal{S}(u)](t),[\mathcal{S}(v)](t)), \quad (3)$$

On the other hand, causality of the system induces the converse inequality. More precisely, since

$$\forall t,\ [min(u,v)](t)\leq u(t),$$

we have

$$\forall t,\ [\mathcal{S}(min(u,v))](t)\leq[\mathcal{S}(u)](t),$$

that is the amount of vehicles out the road section is at any time t greater with u than with $min(u,v)$ as input flow. With similar arguments we have $\forall t,\ [\mathcal{S}(min(u,v))](t)\leq[\mathcal{S}(v)](t)$, and we deduce that

$$[\mathcal{S}(min(u,v))](t)\leq min([\mathcal{S}(u)](t),[\mathcal{S}(v)](t)). \quad (4)$$

Inequalities (3) and (4) satisfied by vehicular traffic flow on a roadway stretch correspond to the first condition (1) defining a min-plus linear system.

[1]Additions or withdrawals of vehicles *via* entry/exit lanes will be taken into account as additional inputs/outputs in the model.

Still considering that no vehicle can disappear along a road section, the amount of vehicles out of the road section with $a+u$ as input flow, is claimed to be equal to the sum of a with the amount of vehicles out of the road section with u as input flow. Furthermore, in the input flow $a+u$ at time t, the a vehicles can be considered to have been added to flow u for a sufficiently long time so that they have already crossed the road section. From these observations, we deduce that a road traffic system satisfies the second condition (2) defining a min-plus linear system.

2.2 Min-plus Representations for Roadway Stretches

We have shown that road sections considered in §2.1 could be studied as min-plus linear systems. Then, they can be represented by their impulse responses (see for example (Gaubert, 1992), (Lahaye, 2000)).

Definition 2.2 (Impulse Response) *Let $\mathcal{S}$ be a min-plus linear system, there exists a unique mapping h, called impulse response, such that $y = \mathcal{S}(u)$ is expressed as:*

$$\forall u, \forall t\,;\ y(t) \triangleq \min_{s\leq t}\{h(s)+u(t-s)\} = (h\otimes u)(t).$$

The system output is nothing but the *inf-convolution* - which plays the role of convolution in min-plus linear systems theory - between its impulse response and the system input u.

In the following, we may rather consider a lower approximation denoted β of an impulse response h. Such an approximation is analogous to the service curve usually used in Network Calculus theory (see (Cruz, 1991a), (Cruz, 1991b), (Boudec and Thiran, 2001)). Considering that:

$$\forall t\,,\quad h(t) \geq \beta(t), \tag{5}$$

we have by isotony of the convolution product ($\otimes$):

$$\forall t\,, y(t) = (h\otimes u)(t) \geq (\beta\otimes u)(t),$$

which means that $(\beta\otimes u)(t)$ is a lower approximation of the system output, that is $(\beta\otimes u)(t)$ gives a minimal flow of vehicles leaving the road section. Such a lower approximation β is used in particular when exact identification of h is not possible. We show in the following lemma that a mapping β' such that $\forall t\,, (\beta'\otimes u)(t) \leq y(t)$ is a lower approximation of the impulse response h of the system.

Lemma 2.1 *Let β' be a mapping such that $\forall u, y \geq \beta'\otimes u$ with $y = h\otimes u$, then we have: $\forall t\,, h(t) \geq \beta'(t)$.*

Proof Let us define the particular signal $\delta(t)$ as:

$$\delta(t) \quad = \quad \begin{cases} 0 & \text{if } t \leq 0 \\ +\infty & \text{otherwise} \end{cases}. \tag{6}$$

Then we easily check that $y = h\otimes u \geq \beta'\otimes u$ implies $h \geq \beta'$ by taking $u = \delta$.

Finally, let us give an interpretation to mappings h and β. An input equal to the signal δ defined by (6), comes down to considering that

- no vehicle is in the system before $t = 0$,
- an infinity of vehicles are available to enter the system as soon as $t > 0$.

Then we have with $u = \delta$

$$\begin{aligned} y(t) = (h\otimes u)(t) &= \min_{s\leq t}\{h(s)+u(t-s)\} \\ &= \min(\underbrace{h(s)+(+\infty)}_{s<t}, \underbrace{h(t)+0}_{s=t}) \\ &= h(t) \end{aligned}$$

So for all t, $h(t)$ can be interpreted as the number of vehicles having crossed the corresponding road section up to time t while an infinity of vehicles could enter the section from time 0. In other words, $h(t)$ can be interpreted as the maximum number of vehicles that can be "served" during the time interval $[0,t]$. Since $h(t) \geq \beta(t)$, $\forall t$, $\beta(t)$ can be interpreted as a lower approximation of the maximum number of vehicles that can cross the section during $[0,t]$.

3 PROPOSED MODELING METHODOLOGY

In this section, a modeling methodology is proposed for roadways as min-plus linear systems. In a first place, we select a particular min-plus linear system to represent any road section. This elementary representation is intended to be sufficiently generic to model various roadway stretches (with specific parameters). These constitute the elementary bricks we shall combine to build models for larger infrastructures. The way these bricks are assembled is explained in a second place.

3.1 A Generic Model for Elementary Road Sections

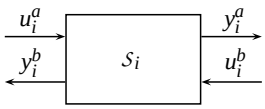


Figure 1: Generic system $\mathcal{S}_i$ with 2 inputs and 2 outputs proposed to represent any road section.

We propose to model any road section as a min-plus linear system with two inputs and two outputs interpreted as the following counters:

- $u_i^a(t)$ denotes the cumulated number of vehicles having entered the section indexed i up to time t,
- $u_i^b(t)$ denotes the cumulated number of vehicles authorized to leave the section i up to time t,
- $y_i^a(t)$ denotes the cumulated number of vehicles having left the section indexed i up to time t,
- $y_i^b(t)$ denotes the cumulated number of vehicles authorized to enter the section i up to time t.

Signals $u_i^b(t)$ and $y_i^b(t)$ will be used to take into account specific phenomena and/or the mutual influences between successive sections. They are intended to enable to model a wide variety of roadway stretches. For examples:

- For a section ending with a traffic light, the input signal $u_i^b(t)$ will be used to traduce the successive light phases.
- Output $y_i^b(t)$ will be suitable to model the effect of a congestion on the upstream sections.

The min-plus linear system $\mathcal{S}_i$ admits the following representation

$$\left\{ \begin{array}{rcl} y_i^a & = & \Theta_i u_i^a \oplus \Sigma_i u_i^b \\ y_i^b & = & \Phi_i u_i^a \oplus \Gamma_i u_i^b \end{array} \right. \tag{7}$$

in which notation $\oplus$ stands for the point-wise minimum of signals, that is $(u \oplus v)(t) = min(u(t), v(t))$, $\forall t$. The impulse responses Θ_i, Σ_i, Φ_i and Γ_i traduce respective influences of the inputs on the two outputs.

3.2 Model for a Succession of Elementary Road Sections

An important feature of linear systems is that they can be cascaded in series, in parallel or put in feedback and then we always get a linear system. In this section, we explain how proposed generic models should be assembled in order to get a model for successive road sections. In particular, we give the representation obtained for two road sections cascaded in series.

Let us consider two successive road sections indexed i and $i+1$, and modeled by min-plus linear systems $\mathcal{S}_i$ and $\mathcal{S}_{i+1}$. They are respectively represented by

$$\left\{ \begin{array}{rcl} y_i^a & = & \Theta_i u_i^a \oplus \Sigma_i u_i^b \\ y_i^b & = & \Phi_i u_i^a \oplus \Gamma_i u_i^b \end{array} \right. , \tag{8}$$

and

$$\left\{ \begin{array}{rcl} y_{i+1}^a & = & \Theta_{i+1} u_{i+1}^a \oplus \Sigma_{i+1} u_{i+1}^b \\ y_{i+1}^b & = & \Phi_{i+1} u_{i+1}^a \oplus \Gamma_{i+1} u_{i+1}^b \end{array} \right. . \tag{9}$$

Cascading $\mathcal{S}_i$ and $\mathcal{S}_{i+1}$ comes down to merging inputs and outputs of $\mathcal{S}_i$ and $\mathcal{S}_{i+1}$ in the following way (see figure (2)):

$$u_{i+1}^a = y_i^a, \tag{10}$$

$$u_i^b = y_{i+1}^b. \tag{11}$$

In other words, we merely consider that:

- the cumulated number of vehicles leaving section i (given by y_i^a) is equal to the cumulated number of vehicles entering section $i+1$ (given by u_{i+1}^a),
- the cumulated number of vehicles authorized to enter section $i+1$ (that is y_{i+1}^b) is equal to the cumulated number of vehicles authorized to leave section i (*i.e.* u_i^b).

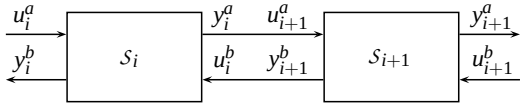


Figure 2: Proposed assembly for $\mathcal{S}_i$ and $\mathcal{S}_{i+1}$ modelling two successive road sections.

Min-plus linear system theory has shown that the resulting system when cascading is also a min-plus linear system (see (Gaubert, 1992) or (Lahaye, 2000)). In the following, we explicit the representation for this system, denoted $\mathcal{S}$, that is the corresponding impulse responses Θ, Σ, Φ and Γ.

Equations (8) and (9) lead to:

$$\left\{ \begin{array}{rcl} y_{i+1}^a & = & \Theta_{i+1} u_{i+1}^a \oplus \Sigma_{i+1} u_{i+1}^b \\ y_i^b & = & \Phi_i u_i^a \oplus \Gamma_i u_i^b \end{array} \right. . \tag{12}$$

From equations (10) and (11) we have:

$$\begin{array}{rcl} u_{i+1}^a & = & y_i^a \\ & = & \Theta_i u_i^a \oplus \Sigma_i u_i^b \\ & = & \Theta_i u_i^a \oplus \Sigma_i y_{i+1}^b \\ & = & \Theta_i u_i^a \oplus \Sigma_i \Phi_{i+1} u_{i+1}^a \oplus \Sigma_i \Gamma_{i+1} u_{i+1}^b \\ & = & (\Sigma_i \Phi_{i+1})^* \Theta_i u_i^a \\ & & \oplus (\Sigma_i \Phi_{i+1})^* \Sigma_i \Gamma_{i+1} u_{i+1}^b \end{array} \tag{13}$$

The star notation a^* (often referred to as "Kleene star operation") stands for $\bigoplus_{n \in \mathbb{N}} \underbrace{a \otimes a \ldots \otimes a}_{\text{n times}}$. The last equality gives the least solution to previous implicit equation (see (Baccelli et al., 1992, §4.5)). Selecting the least solution means that we are interested in the earliest functioning of the system $\mathcal{S}$. On the other hand, we have:

$$\begin{aligned} u_i^b &= y_{i+1}^b \\ &= \Phi_{i+1}u_{i+1}^a \oplus \Gamma_{i+1}u_{i+1}^b \\ &= \Phi_{i+1}y_i^a \oplus \Gamma_{i+1}u_{i+1}^b \\ &= \Phi_{i+1}\Theta_i u_i^a \oplus \Phi_{i+1}\Sigma_i u_i^b \oplus \Gamma_{i+1}u_{i+1}^b \\ &= (\Phi_{i+1}\Sigma_i)^*\Phi_{i+1}\Theta_i u_i^a \\ &\quad \oplus (\Phi_{i+1}\Sigma_i)^*\Gamma_{i+1}u_{i+1}^b \end{aligned} \tag{14}$$

Substituting signals u_{i+1}^a and u_i^b by (13) and (14) in equations (12) leads to obtain outputs of $\mathcal{S}$ (that is y_{i+1}^a, y_i^b) in response to its inputs (that is u_i^a, u_{i+1}^b):

$$\left\{ \begin{array}{rl} y_{i+1}^a = & \Theta_{i+1}\left[(\Sigma_i\Phi_{i+1})^*\Theta_i u_i^a \oplus (\Sigma_i\Phi_{i+1})^*\Sigma_i\Gamma_{i+1}u_{i+1}^b\right] \\ & \oplus \Sigma_{i+1}u_{i+1}^b \\ y_i^b = & \Phi_i u_i^a \\ & \oplus \Gamma_i\left[(\Phi_{i+1}\Sigma_i)^*\Phi_{i+1}\Theta_i u_i^a \oplus (\Phi_{i+1}\Sigma_i)^*\Gamma_{i+1}u_{i+1}^b\right] \end{array} \right.$$

$$\Leftrightarrow$$

$$\left\{ \begin{array}{rl} y_{i+1}^a = & (\Sigma_i\Phi_{i+1})^*\Theta_{i+1}\Theta_i u_i^a \\ & \oplus \left[\Sigma_{i+1} \oplus (\Sigma_i\Phi_{i+1})^*\Theta_{i+1}\Sigma_i\Gamma_{i+1}\right]u_{i+1}^b \\ y_i^b = & \left[\Phi_i \oplus (\Phi_{i+1}\Sigma_i)^*\Phi_{i+1}\Theta_i\Gamma_i\right]u_i^a \\ & \oplus (\Phi_{i+1}\Sigma_i)^*\Gamma_{i+1}\Gamma_i u_{i+1}^b \end{array} \right.$$

4 EXAMPLES

We have shown in section 2 how road traffic can be studied as a min-plus linear system. In this section, we apply the modeling methodology proposed at section 3 to model and simulate two kinds of elementary roadway stretches.

4.1 Elementary Stretch of a Roadway

As a first example we consider a simple stretch of roadway without any facilities: no traffic light, no intersection, no entry and no exit lanes... Furthermore we assume that the stream of vehicles on the considered road section is not affected by upstream and downstream traffic. According to the methodology proposed in section 3, the road section is described as a min-plus linear system $\mathcal{S}_1$ represented by:

$$\left\{ \begin{array}{rcl} y_1^a &=& \Theta_1 u_1^a \oplus \Sigma_1 u_1^b \\ y_1^b &=& \Phi_1 u_1^a \oplus \Gamma_1 u_1^b \end{array} \right.$$

with

$$\begin{array}{ll} \Theta_1 = h_1 & \Sigma_1 = \varepsilon \\ \Phi_1 = \varepsilon & \Gamma_1 = \varepsilon \end{array}$$

in which ε denotes the "null signal" (in this case the "null impulse response"), with respect to the additive law $\oplus$ (corresponding to the pointwise min). That is:

$$\forall t, \varepsilon(t) = +\infty$$

and we have for all signal u

$$\forall t, \varepsilon(t) \oplus u(t) = u(t) \oplus \varepsilon(t) = min(u(t), +\infty) = u(t).$$

And so we simply have

$$y_1^a = h_1 \otimes u_1^a.$$

As a reminder, $h_1(t)$ (respectively $\beta_1(t)$) denotes the maximal number (resp. a lower approximation of the maximal number) of vehicles that can cross the corresponding road section up to time t while an infinity of vehicles could enter the section from time 0. In future works such models h_1 and β_1 are intended to be obtained through an identification procedure using road traffic counting experiments. Previous works (see for example (Menguy et al., 2000)) have investigated such identification problems for linear systems on idempotent semi-rings (such as the min-plus algebra). By anticipation, we consider here a mapping β_1 as defined on figure 3. The latency (equal to T seconds) corresponds to time spent to cross the road section for an isolated vehicle (not slowed down by other vehicles). The slopes at $t > T$ are then related to the flow of vehicles on the road section knowing that $\beta_1(t)$ vehicles have passed the section up to time t. The chosen shape for β_1 take care of the fact that vehicles run slower in relation to the density of traffic.

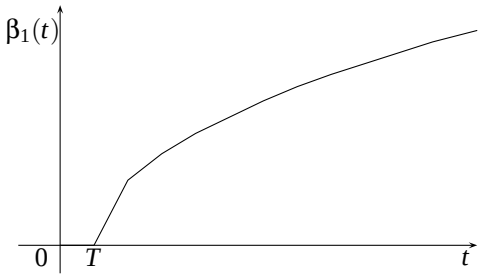


Figure 3: Mapping β_1 used to model a simple roadway stretch.

Thanks to the considered min-plus linear representation, traffic can be simulated, notably by using a C++ library developed by the COINC research group (COINC, 2009). This software enables to define signals as functions composed of segments (such as β_1 in figure 3) and implement the expected operations on signals: min-plus convolution ($\otimes$), pointwise min ($\oplus$), Kleene star operation ($*$), ... We here have used the library to compute the signal $\beta_1 \otimes u_1^a \geq y_1^a$ with u_1^a defined as ramp signal with slope rate r. This means to considering that vehicles enter indefinitely the section from time 0 with a constant input flow r (*i.e.* r vehicles enter per second). Doing so with r varying

from a value lower than the asymptotic rate of β_1 to a value greater to it (see figure 4), we expect to study the behaviour for all traffic conditions.

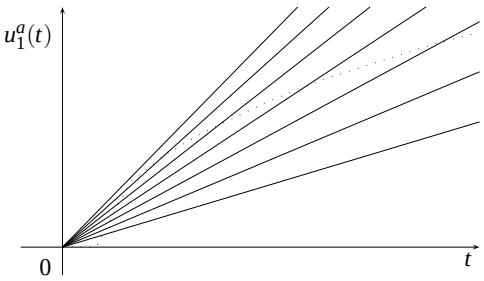


Figure 4: Input signals u_1^q considered for the simulations.

For each simulation (with a given value of r), we have computed for all $t \in [0,200]$

- the so-called "backlog", that is the amount of vehicles on the section at time t, given by

$$u_1(t) - y_1(t)$$

- the "virtual delay", that is the travel time for vehicle(s) entered in the section at time t if they don't overtake vehicles entered before. The virtual delay is given by

$$\inf\{\tau \geq 0 : u_1(t) \leq y_1(t+\tau)\}.$$

These values are respectively proportional to the vehicles density K and the flow Q. We then obtain the *fundamental diagram* of figure 5.This diagram gives the relation between the flow and the vehicles density for the car traffic on a road. It has been observed empirically and derived theoretically in the case of a unique road or a regular system of roads (see for example (Helbing, 2001)).

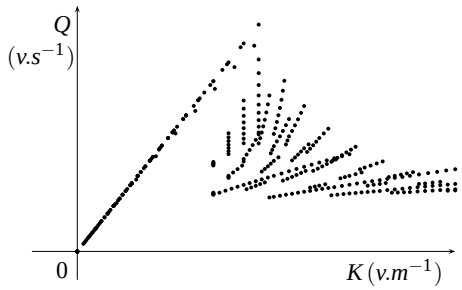


Figure 5: Fundamental diagram obtained for a simple road section.

Let us mention that with this model if signal u_1^q is such that $\lim_{t\to\infty} u_1^q(t) - \beta_1(t) = +\infty$, then since $y_1^q(t) = \min_{s\leq t}(\beta_1(t-s) + u_1^q(s)) \leq \beta_1(t)$ we have $\lim_{t\to\infty} u_1^q(t) - y_1^q(t) \geq \lim_{t\to\infty} u_1^q(t) - \beta_1(t) = +\infty$. This means that if the arrival of vehicles on the road exceeds its service capacity then the section will asymptotically contain an infinity of vehicles. The next example notably shows how to take into account the intrinsic limited capacity of a section.

4.2 Succesion of Elementary Stretches with Limited Capacities

We now consider two successive road sections as represented on figure 6. These roadway stretches modeled by β_1 and β_2 are also supposed to be without any facilities (no traffic light, ...), but W_1 and W_2 have been added to limit their capacities. More precisely, the maximum amount of vehicles that section 1 can contain is supposed to correspond to the integer W_1. We then have

$$\forall t, u_1^q(t) = min(u(t), y_1^q(t) + W_1),$$

in which $u_1^q(t)$ denotes the amount of vehicles likely to enter section 1 up to time t. This leads to

$$\forall t, u_1^q(t) \leq y_1^q(t) + W_1 \;\Leftrightarrow\; u_1^q(t) - y_1^q(t) \leq W_1$$

which shows that the number of vehicles on the section given by $u_1^q(t) - y_1^q(t)$ is then well bounded by W_1.
Referring to equations (7) defining the generic model, we deduce for $i = 1,2$ that

$$\begin{array}{ll} \Theta_i = h_i & \Sigma_i = e \\ \Phi_i = W_i h_i & \Gamma_i = W_i \end{array}.$$

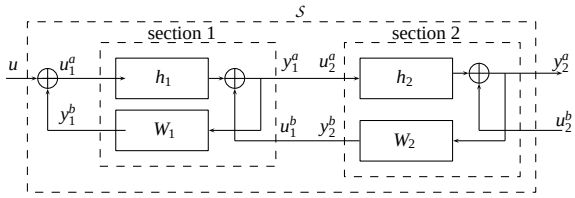


Figure 6: Two successive road sections.

From results detailed in section 3.2, we can compute the representation for the system resulting from section 1 and 2 in cascade. Defining two mappings β_1 and β_2 comparable (but different slopes) as that of figure 3, we have simulated the system as in section 4.1. We then obtain two fundamental diagrams:

- diagram of figure 7 obtained for flows between u and y_2^q, with $u(t)$ denoting the amount of vehicles which have been candidates for entering section 1 up to time t;

- diagram of figure 8 obtained for flows between u_1^q and y_2^q, with $u_1^q(t)$ denoting the amount of vehicles having entered section 1 up to time t.

Figure 7: Fundamental diagram for two successive sections with limited capacities (flows between u and y_2^q).

Figure 8: Fundamental diagram for two successive sections with limited capacities (flows between u_1^q and y_2^q).

5 DISCUSSION ON THE PROPOSED (MIN,+) MODEL

For about fifty years, mathematical description of traffic flow has been a lively subject of research and debate for traffic engineers. This has resulted in a broad scope of models describing different aspects of traffic flow operations which can be classified according to various criteria: level of detail, application area and scale of application, deductive or inductive description of phenomena,...(see for example the survey (Hoogendoorn and Bovy, 2001)). In this section, we discuss the location of the proposed model in these classifications.

Traffic flow can be described either by considering the time-space behavior of individual drivers under the influence of vehicles in their proximity (microscopic models), the behaviour of drivers without explicitly distinguishing their time-space behavior (mesoscopic models), or from the viewpoint of the collective vehicular flow (macroscopic models). According to the level-of-detail, variables have different natures. In a microscopic model variables describe individually behaviors and interactions of the systems entities (i.e. vehicles and drivers). In macroscopic flow models, the traffic stream is represented in an aggregate manner using characteristics as flow-rate, density, and velocity. In our model, we manipulate functions which count every vehicle (from which we have derived aggregated characteristics such as flow-rate and density), but their behavior is not necessarily described individually through the approximation β of the impulse response (unless the road section is sized such that it contains only one vehicle). In that sense, our model could be considered as an intermediate approach which, according to the size of the modeled sections, fluctuates between microscopic and macroscopic approaches. Our description of observed phenomena is also somewhat intermediate. In fact, the choice of β and the limitation of capacity on sections come under a deductive approach of phenomena which are known or which can be guessed. But the mapping β should be the fitted thanks to an inductive approach, that is using input/output data from real systems.

Although it has only be used to simulate traffic flow in the present paper, we expect that analytical results can be derived from the proposed model. In fact, for more than two decades a new system theory has been developed for systems linear over idempotent semi-rings (such as min-plus algebra): numerous results have been proposed for performance evaluation, control,... (see (Cohen, 2006) for a recent survey). In future works, some of these results should be applied/adapted to study vehicular traffic flow.

In this paper, we have considered elementary stretches of roadways. On the one hand, we expect that the chosen framework is sufficiently generic to model a wide variety of more complex road sections (with or without entry/exit lane, intersections, ...). On the other hand, the way to aggregate elementary models is simple enough to consider that models for large infrastructures can be obtained.

6 CONCLUSIONS

We have proposed a modeling method for traffic flow. Each road section is modeled by its impulse response in min-plus algebra. This model has been used to simulate and derive the fundamental diagram for elementary road stretches.

Future works will concern an identification method to build an approximation β of the impulse response from real road traffic data. We also plan to study more

complex roadway stretches (for example phenomena associated to a traffic light), and to adapt existing results from linear system theory over min-plus algebraic to derive analytical results on traffic flow.

REFERENCES

Baccelli, F., Cohen, G., Olsder, G. J., and Quadrat, J. P. (1992). *Synchronization and Linearity*. Wiley.

Boudec, J. Y. L. and Thiran, P. (2001). *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Springer.

Cohen, G. (2006). *A Tour of Systems with the Max-Plus Flavor*, volume 341 of *Lecture Notes in Control and Information Sciences*, chapter Positive Systems, pages 19–24.

COINC, R.-G. (2009). Computational issues in network calculus. http://perso.bretagne.ens-cachan.fr/˜ bouillar/coinc/.

Cruz, R. L. (1991a). A calculus for network delay, part I: Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–131.

Cruz, R. L. (1991b). A calculus for network delay, part II: Network analysis. *IEEE Transactions on Information Theory*, 37(1):132–141.

Farhi, N., Goursat, M., and Quadrat, J. P. (2007). Road traffic models using petri nets and minplus algebra. In *Proceedings of the Traffic and Granular Flow Conference*, Orsay, Paris.

Gaubert, S. (1992). *Théorie des systèmes linéaires dans les dioides. Thèse*. PhD thesis, Ecole des Mines de Paris.

Helbing, D. (1996). Gas-kinetic derivation of Navier-Stokes-like traffic equations. *Physical Review E*, 53(3):2366–2381.

Helbing, D. (2001). Traffic and related self-driven many-particle systems. *Reviews of modern physics*, 73:1067–1141.

Hoogendoorn, S. P. and Bovy, P. H. L. (2001). State-of-the-art of Vehicular Traffic Flow Modelling. *Journal of Systems and Control Engineering*, 215(4):283–303.

Lahaye, S. (2000). *Contribution à l'étude des systèmes linéaires non stationnaires dans l'algèbre des dioides*. PhD thesis, Université d'Angers.

Lolito, P., Mancinelli, E., and Quadrat, J. P. (2005). A Minplus Derivation of the Fundamental Car-Traffic Law. *IEEE Transactions on Automatic Control*, 50(5):699–705.

Menguy, E., Boimond, J. L., Hardouin, L., and Ferrier, J. L. (2000). A First Step Towards Adaptative Control for Linear Systems in Max-Plus Algebra. *Discrete Event Dynamic Systems: Theory and Applications*, 10(1):347–367.

Nagel, K. and Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. *Journal de Physique I France*, 2(2):2221–2229.

GENERATING HIGH-SPEED THREE-DIMENSIONAL DYNAMIC QUADRUPED WALKING USING AN EVOLUTIONARY SEARCH

Di He, Qining Wang, Chunxia Rong and Guangming Xie
Intelligent Control Laboratory, College of Engineering, Peking University, Beijing 100871, China
xiegming@pku.edu.cn

Keywords: Quadruped walking, Three-dimensional gaits, Evolutionary search, Legged robots, Locomotion.

Abstract: This paper presents an evolutionary computation approach to generate three-dimensional fast forward gaits for quadruped robots with three motor-driven joints on each limb. We use linear constraints to reduce the high-dimensional space of parameters in order to generate the speed effectively. Real robot experiments show that the evolutionary approach is effective in developing quadruped gaits. Satisfactory results are obtained in about an hour by the autonomous learning process, which starts with a set of hand-tuned parameters.

1 INTRODUCTION

Over the past years, plenty of publications have been presented in the biomechanics literature which explained and compared the dynamics of different high-speed gaits including gallop, canter, bound, and fast trot (e.g. (Alexander and Jayes, 1983), (Alexander et al., 1980)). To study and implement legged locomotion, various robot systems that achieve high-speed and efficient walking gaits have been created (e.g. (Holmes et al., 2006), (Raibert, 1986), (Collins et al., 2005)). Since the walking speed of the robots is one of the most important factors in determining the success of a team in the RoboCup Four-Legged competitions with the standard platform using Sony AIBO ERS-7 robots, there has been significant incentive in the RoboCup community to develop gaits with better performance on the speed (e.g. (Hornby et al., 1999), (Röfer et al., 2004), (Quinlan et al., 2003)).

Theoretically, high-speed gait design can be achieved by dynamics analysis, if it can be simplified into dynamics models. However, quadruped robots' dynamics state are complex with the ground constraints. Especially when dealing with the whole robot, the problem becomes nonlinear and high-dimensional. Therefore, current methods that dealt with gait optimization often resort to Inverse Kinematics Model, which convert the optimizing problem into a gait locus design (e.g. (Quinlan et al., 2003), (Röfer et al., 2004), (Röfer et al., 2005), (Rong et al., 2009)). In previous work, researchers have designed several locus shapes, e.g. rectangular, elliptical, trapezoidal and three-dimensional polygon to describe the walking patterns for AIBO robots. With different machine learning algorithms, the gait locus can be improved. For example, the sharP-KUngfu Team proposed an Adaptive Particle Swarm Optimization (APSO) based approach to generate fast two-dimensional gaits and reaches a speed of $425mm/s$ (Rong et al., 2009). The team from the University Newcastle generated fast gaits which is $420mm/s$ using Genetic Algorithm (GA) and plane loci of arbitrary shape (Quinlan et al., 2003). However, these methods only deal with the generation of two-dimensional quadruped walking. The three-dimensional walking gaits are more close to real walking patterns of the quadruped animals. Only a few studies have been made in the generation of three-dimensional walking gaits. For example, the German Team created a flexible gait implementation that controls the feet on a path described by a three-dimensional polygon and get a speed of $451mm/s$ finally (Röfer et al., 2005). The results indicated that gait optimization with a three-dimensional polygon can obtain stable gaits with higher walking speed. However, as the complexity of the polygon increases, the learning procedure is time consuming and may damage the motor-driven joints of the robots during the optimization process of the high dimensional parameters.

As a matter of fact, the performance of optimization is not the same by using different gait locus shapes or machine learning algorithms. Generally, optimizing with a model of more degrees of freedom

is likely to generate a better gait, but it takes more time. And an algorithm which has a better global search capability always converges slowly. In this paper, we use three-dimensional polygon as the gait model for the locus shape. To solve the problem of slow convergence of the optimization using three-dimensional polygon, we propose a method based on linear constraints to reduce the degrees of freedom of the model. The whole learning process is running automatically by the robot with onboard processor. In real robot experiments, we achieved an effective gait which speed is higher than the previous known gaits, using AIBO as the test platform.

The remainder of this paper is organized as follows. Section II introduces the parameterization and kinematics for the Sony AIBO ERS-7 robot platform, and the general methods for quadruped gait planning. Section III presents our evolutionary search based on linear constraints. Section IV specifies the evaluation of gait optimization based on the proposed evolution and others. Section V presents the conclusion and discussion.

2 DYNAMIC MODEL FOR AIBO ROBOT

2.1 Inverse Kinematics Model

The high-level parameters that we adopt to represent the gait need to be transferred to joint angles of legs before they can be implemented by the robot. An inverse kinematics model can be used to solve this problem. For a linked structure with several straight parts connecting with each other, the position of the end of this structure relative to the starting point can be decided by all angles of linked parts and only one position results from the same angle values. The definition of the kinematics model is the process of calculating the position of the end of a linked structure when given the angles and length of all linked parts.

For the AIBO robots, Given the position of the end of the structure, inverse kinematics calculates out what angles the joints need to be in to reach that end point. In this study, the inverse kinematics is used to calculate necessary joint angles to reach the paw position determined by gait parameters. Fig. 1 shows the inverse kinematics model and the coordinates for AIBO.

The shoulder or hip joint is the origin of the coordinate system. l_1 is the length of the upper limb, while l_2 is the length of the lower limb. Paw position is represented by point (x, y, z). The figures and equations here only give the view and algorithm to get the solution for left fore leg of robot. According to the symmetrical characteristic of legs, all other legs can use the same equations with some signs changing.

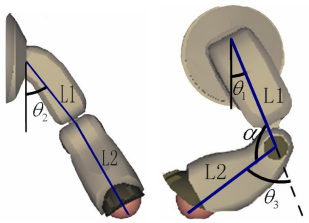


Figure 1: The inverse kinematics model and coordinates for Aibo. (a) is the front view of left fore leg. (b) is the side view of left fore leg.

The following equations shows the inverse kinematics model:

$$\begin{aligned} x &= l_2\cos\theta_1\sin\theta_3 + l_2\sin\theta_1\cos\theta_2\cos\theta_3 + l_1\sin\theta_1\cos\theta_2 \\ y &= l_1\sin\theta_2 + l_2\sin\theta_2\cos\theta_3 \\ z &= l_2\sin\theta_1\sin\theta_3 - l_2\cos\theta_1\cos\theta_2\cos\theta_3 - l_1\cos\theta_1\cos\theta_2 \end{aligned} \quad (1)$$

The inverse kinematics equation to get θ_1, θ_2, θ_3 by the already known paw position (x, y, z) is as follows:

$$\begin{aligned} \theta_3 &= \cos^{-1}\frac{x^2+y^2+z^2-l_1^2-l_2^2}{2l_1l_2} \\ \theta_2 &= \sin^{-1}\frac{y}{l_2\cos\theta_3+l_1} \\ \theta_1 &= -\tan^{-1}\frac{a}{b} \pm \cos -1\frac{x}{a^2+b^2} \end{aligned} \quad (2)$$

where $a = l_2\sin\theta_3$, $b = -l_2\cos\theta_2\cos\theta_3 - l_1\cos\theta_2$.

One problem of the inverse kinematics is that it always has more than one solution for the same end point position. However, as to AIBO, only one solution is feasible due to the restriction on the joint structure. As a result, when using inverse kinematics to calculate joint angles, it is necessary to take joint structure limitation into consideration to get the right solution. Otherwise, it will possibly cause some physical damage to the robot platform.

2.2 Control Parameters

Before we run the learning gait procedure, the control parameters representing a gait need to be decided. For the stance parameters, as shown in Section 2.1, point (x, y, z) can be translated into $(\theta_1, \theta_2, \theta_3)$. Therefore, we use coordinates of the paw positions relative to the shoulders to describe the posture, which is called *FootZeroPosition*. As to locus, we choose an arbitrary three-dimensional polygon with n vertices (P_1, P_2, ..., P_n) as the shape. Additionally, n timing parameters which we call *p_length*, are needed to specify the amount of time needed for the foot to travel

Figure 2: The end of a foot travels as a 3-dimensional polygon.

from vertex P_i to P_{i+1}, where $1 < i < n$, and from P_n to P_1, the total time needs for one cycle is the step length, which we assume here is an integral number, in 0.008 second units. In this way, a total number of parameters for a single leg of $4n+4$. And we suppose the two diagonally feet are traveling at the same time while the other two delay for half a period. Thus, the total number of parameters for the AIBO robot during the proposed gait optimization process is $8n+8$.

2.3 Evolution of the Parameters

We can calculate the walking posture from point to point and moment to moment by the model above. However, because of the constraint of the ground, the end of the foot in fact will not travel exactly as the polygons we design, the relation between gait parameters and speed is impossible to acquire, and there is no sufficiently accurate simulator for AIBO due to the dynamics complexity. As a result, we have to perform the learning procedure on real robots.

In order to automatically acquiring speed for each parameter set, the robot has to be able to localize itself. Since the low resolution of AIBO's camera and limited processing ability, it is faster and more accurate to detect black-white edge than other things. Thus we use a white board with parallel black bars on the field for AIBO to localize (see Fig. 3).

During evaluation procedure, the robot walks to a fixed initial position relative to the board, then loads the parameter set needed to be evaluated, walks for a fixed time, e.g. $6s$, stops and calculates the forward speed. After that, the robot walked back and started another trial. The total time for testing one set of parameter is usually less than $15s$.

Figure 3: The white board with parallel black bars for body adjusting of the AIBO robots.

3 GAIT OPTIMIZATION WITH LINEAR CONSTRAINTS

Since the paw of one foot travels a smooth curve in space, the more complex the polygon that we use to describe the locus shape, the more close the walking gait to that of the real robot. However, in practice, the optimization with a higher dimensional model is more easily to damage the robot in the gait evolution and is time-consuming. Here, we propose a new strategy which reduces the degree of freedom of the model by linear constraints.

Assume that the paw of one foot travels as a three-dimensional polygon with 2n-vertices Q_1, Q_2, $\ldots$ Q_{2n}. The paw travels from Q_i to Q_{i+1} while $i = 1,2,\ldots,2n-1$ and Q_{2n} to Q_1. Let t_1, t_2, …, t_{2n} be the run-through-time for the edges relative to the time for the whole polygon, which is under the constraint:

$$\sum_{i=1}^{2n} t_i = 1 \tag{3}$$

Given the polygons for the front and rear legs respectively, it produces a $16n$ dimensional space of the locus for the whole robot. Then the linear constraint is as follows:

$$\begin{cases} Q_{2k+1} = (Q_{2k} + Q_{2k+2})/2, & k = 1,2,\ldots,n-1 \\ Q_1 = (Q_2 + Q_{2n})/2, & \\ t_{2k-1} = t_{2k}, & k = 1,2,\ldots,n. \end{cases} \tag{4}$$

Under the linear constraint, the dimension of locus reduces by half. Furthermore, since the vertex Q_{2k+1} is the midpoint of Q_{2k} and Q_{2k+2}, a $2n$ vertex polygon changes into an n vertex polygon in fact. It means the gait optimization can begin with a simple locus shape first, such as a three-dimensional polygon with 4-vertices. When the speed approaches to a certain level, change the locus shape into a polygon with 8-vertices by linear interpolation and continue the procedure.

The optimization process can be divided into two parts. In the first part, the optimization process focuses on global search by using linear constraints. When approaches the optimal solution, the optimization process moves to the second part, where the linear constraints will be removed and the local search

Table 1: Control parameters in gait evolution.

Parameter	Description
foot zero position	relative position of paws and shoulders
points of the polygon	relative position of the locus and Foot Zero Position
p-length	run-through-time for the edge relative to the time for the whole polygon
step length	time for one complete step in 0.008 second units

will be emphasized. We can apply different optimization approaches, e.g. APSO, GA, to different parts of the optimization process.

4 EXPERIMENTAL RESULTS

Using the method described above, we carried out two separate experiments and evaluate the walking results. In the first experiment, since different machine learning algorithms perform differently during optimizations, we evaluate the performance of using GA or APSO. Similar to (Rong et al., 2009), the inertial weight of APSO is determined by the equation:

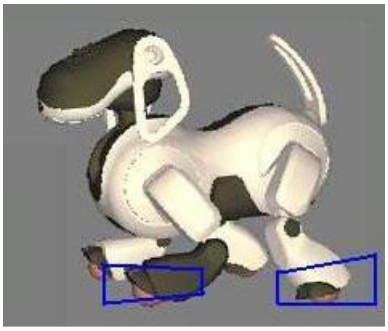
$$\begin{aligned} w &= 1.2 - 0.02 \times k, \ (k \leq 10) \\ w &= 1 - 0.085 \times (k-10), \ (10 < k \leq 20) \\ w &= 0.15 - 0.03 \times (k-20), \ (20 < k \leq 25) \\ w &= 0, \ (k > 25) \end{aligned} \quad (5)$$

where k is the iteration. Three-dimensional polygon with 4-vertices (40-dimensional search space) is chose as the locus shape for the low-dimensional search section. We begin the optimization with a hand-tuned gait which is $250mm/s$ as initial state. After the speed rising to about $400mm/s$, we change the optimized locus into an 8-vertices polygon by linear interpolation and continue the evolution. The process is shown in Fig. 4.

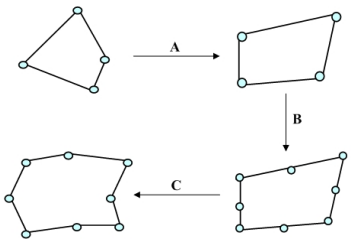
For both of the optimization algorithms, the number of individuals for one population is chosen to be 10. And in the genetic algorithm we use five better individuals to generate the other five by crossover operator and mutation operator.

Fig. 5 shows the best result of each generation in different strategies. We can see that the strategy of using APSO first and optimizing with GA after linear interpolation achieved better result than the other three. Moreover, We can note that there are both advantages and disadvantages comparing these experiments with different algorithms. Fig. 5 indicate that APSO can find the optimal solution area during the early period of the global search.

The optimized speed of the $15th$ generation reaches about $400mm/s$. However, the ability of local search is weak and the convergence is relatively



(a)



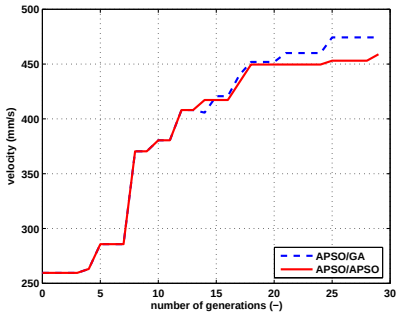
(b)

Figure 4: (a) shows the initial locus for the robot, (b) shows the optimization process: in procedure *A*, we optimize the locus shape of 8-vertices polygon with linear constraint, which is equal to optimize a gait of 4-vertices polygon. In procedure *B*, we change the locus shape by linear interpolation. In procedure *C*, we optimize with the high dimensional locus shape of 8-vertices polygon.

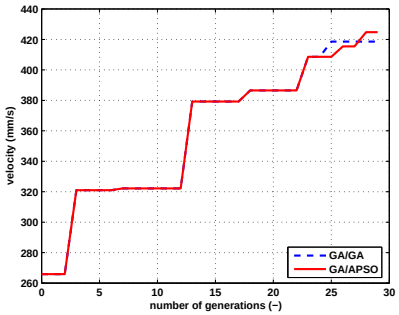
slow. Contrarily, GA grows slowly during the early period of global search, the optimized speed reaches about $400mm/s$ until the optimization process runs to the $25th$ generation. However, if given a relatively optimal solution, GA can find the best result within a short time period. Thus, we select the strategy which integrates APSO and GA to inherit the advantages of them.

In the second experiment, we compare our method with the existing methods (see Fig. 6).

It indicates that during learning, GA performs slowly with speed result $403mm/s$ after 30 generations. APSO grows fast in the first 13 generations, but grows slow then and reaches a speed of $435mm/s$



(a)



(b)

Figure 5: The velocity of the best gait from each generation by using different strategies. e.g. GA/APSO (the red line in (b)) means using GA to optimize the 4-vertices polygon and optimize the 8-vertices polygon with APSO after the linear interpolation.

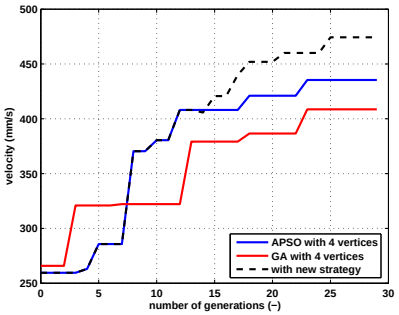
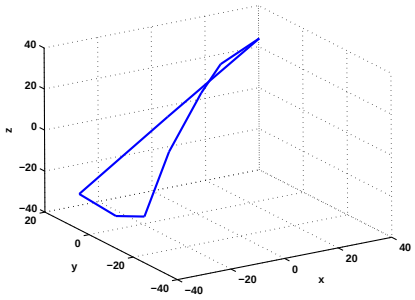


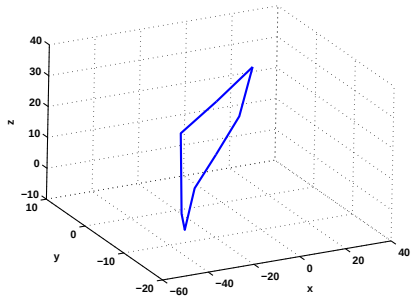
Figure 6: The velocity of the best gait from each generation in real robot experiments.

at last. Our new search strategy achieves better result than the other two, and the speed reaches to $478 mm/s$. It suggests that more dimensions of the search space can get a better gait than the less one. And our strategy avoids the slow learning process. Fig. 7 and Fig. 8 show the locus of the paws and the walking performance of the real robot respectively.

The final speed of the optimized quadruped gait is faster the the existing results using AIBO platform.



(a)



(b)

Figure 7: locus.

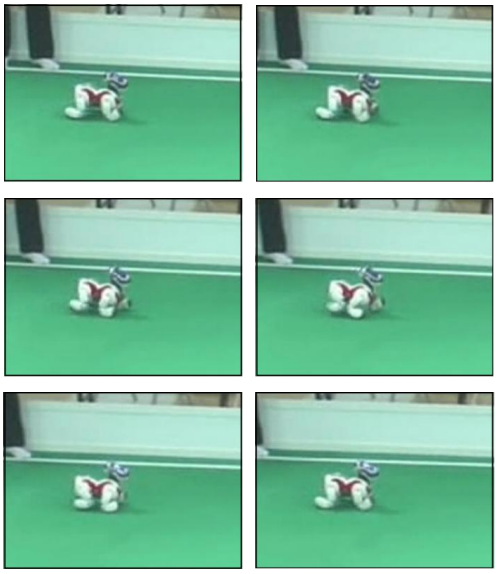


Figure 8: A sequence of photos captured during autonomous walking of the AIBO robot on carpet.

5 CONCLUSIONS

In this paper, we have demonstrated a novel evolutionary computation approach to generate three-dimensional quadruped fast forward gaits using the AIBO robot platform. Our method was easily coded

and computationally inexpensive. Moreover, by linear constraints, the evolution converged extremely fast and the training time was largely reduced. It is an essential advantage for physical robot learning, minimizing possible damage to the robot. It reduced the human work as well as generating evolutional results varied a lot in different experiences. Through experiments which took about 60 minutes each, we achieved several high performance sets of gait parameters which differ a lot from each other. The proposed method generated a speed of $478mm/s$ which is faster than the previous known gaits.

One of the useful aspects of the proposed method is that the high-dimensional parameter set is optimized in an effective way. In order to reduce the dimension, we approach a linear constraint for the parameters. Under this constraint, the locus can be optimized fast. Another contribution of our method is the combination of the two algorithms, GA and APSO. The result shows that the strategy has better global searching capability and local searching capability than using each algorithm only.

In the future, we will compare different high-performance gait parameters and analyze the dynamics model of the robot to obtain further understanding of the relation between parameter and its performance. In this study, we find that the gait actually executed by the robot differs significantly from the one that we design. There are several possible reasons. The most important one is the interaction with environment prevents the implement of some strokes of robot legs. Although with learning approach, factors that cause the difference between actual gait and planned gait do not have to be taken into consideration. However, we assume that if the planned gait and actual gait can conform to each other, AIBO will walk more stable with high speed. In order to solve the problem, the analysis of dynamics between the robot and the environment is necessary. In the gait learning procedure of current study, we only evolve fast forward gait and choose forward speed as the fitness. Later on, we will try to learn effective gaits in other directions, for example, gaits for walking backward, sideward and turning. We also consider exploring optimal omni-directional gaits. With gaits working well at all directions, robots will be able to perform more flexibly and reliably.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the contribution of the team members of the sharP-KUngfu Robot Team. Part of the source code and videos can be found on the web page http://www.mech.pku.edu.cn/robot/fourleg/

REFERENCES

R. M. Alexander, A. S. Jayes, A dynamic similarity hypothesis for the gaits of quadrupedal mammals, *J. Zoology*, vol. 201, pp. 135-152, 1983.

R. M. Alexander, A. S. Jayes, R. F. Ker, Estimates of energy cost for quadrupedal running gaits, *J. Zoology*, vol. 190, pp. 155-192, 1980.

S. Collins, A. Ruina, R. Tedrake, M. Wisse, Efficient bipedal robots based on passive dynamic walkers, *Science*, vol. 307, 2005, pp. 1082-1085.

P. Holmes, R. J. Full, D. Koditschek, J. Guckenheimer, The dynamics of legged locomotion: Models, analyses, and challenges, *SIAM Review*, vol. 48, no. 2, 2006, pp. 207-304.

G. S. Hornby, M. Fujita, S. Takamura, T. Yamamoto, O. Hanagata, Autonomous evolution of gaits with the Sony quadruped robot, *Proceedings of the Genetic and Evolutionary Computation Conference*, vol. 2, 1999, pp. 1297-1304.

M. J. Quinlan, S. K. Chalup, R. H. Middleton. Techniques for improving vision and locomotion on the Sony AIBO robot. *Proceedings of the 2003 Australasian Conference on Robotics and Automation*, 2003.

M. J. Quinlan, Nubots team robocup 2005. Technical report, 2005.

M. H. Raibert, *Legged robots that balance*, MIT Press, Cambridge, 1986.

C. Reynolds, Flocks, herds, and schools: A distributed behavioral model, *Comp. Graph.*, vol. 21, no. 4, pp. 25-34, 1987.

C. Rong, Q. Wang, Y. Huang, G. Xie, L. Wang, "Autonomous evolution of high speed quadruped gaits using particle swarm optimization", *Lecture Notes in Artificial Intelligence*, 5399, Springer-Verlag Berlin Heidelberg, 2009.

T. Röfer, H. D. Burkhard, U. Düffert, J. Hoffmann, D. Göhring, M. Jüngel, M. Lötzsch, O. v. Stryk, R. Brunn, M. Kallnik, M. Kunz, S. Petters, M. Risler, M. Stelzer, I. Dahm, M. Wachter, K. Engel, A.Osterhues, C. Schumann, J. Ziegler, GermanTeam RoboCup 2004. Technical report, 2004.

T. Röfer, H. D. Burkhard, U. Düffert, J. Hoffmann, D. Göhring, M. Jüngel, M. Lötzsch, O. v. Stryk, R. Brunn, M. Kallnik, M. Kunz, S. Petters, M. Risler, M. Stelzer, I. Dahm, M. Wachter, K. Engel, A. Osterhues, C. Schumann, J. Ziegler, GermanTeam RoboCup 2005. Technical report, 2005.

MODEL-ORDER REDUCTION OF SINGULARLY PERTURBED SYSTEMS BASED ON ARTIFICIAL NEURAL ESTIMATION AND LMI-BASED TRANSFORMATION

Othman M-K. Alsmadi, Za'er S. Abo-Hammour
Department of Electrical Engineering, University of Jordan, Amman, Jordan
Department of Mechatronics Engineering, University of Jordan, Amman, Jordan
othman_mk@yahoo.com, zaer_hmr@yahoo.com

Mohammad S. Saraireh
Department of Computer Engineering, Mutah University, Karak, Jordan
srayreh_2000@yahoo.com

Keywords: Model order reduction, System transformation, Artificial neural networks, Linear matrix inequality (LMI), Singular perturbation.

Abstract: A new method for model order reduction with eigenvalue preservation is presented in this paper. The new technique is formulated based on the system state matrix transformation which preserves the system eigenvalues and is accomplished using an artificial neural network training. A linear matrix inequality (LMI) numerical algorithm technique is used to obtain the complete system transformation. Model order reduction is then obtained utilizing the singular perturbation method. Simulation results show that the LMI-based transformed reduced model order is superior to other reduction methods.

1 INTRODUCTION

The objective of any control system is to obtain a desired response. In order to achieve this objective, a dynamical model is usually developed based on a set of differential equations (Franklin, 1994). The obtained mathematical model may have a certain parameter, called perturbation, that has a little effect on the system performance (Kokotovic, 1986) (Zhou, 2009). Neglecting this parameter results in simplifying the order of the designed controller based on reducing the system model order. A reduced model order can be obtained by neglecting the fast dynamics (i.e., non-dominant eigenvalues) of the system and focusing on the slow dynamics (i.e., dominant eigenvalues). This method is referred to as singular perturbation. Simplification and reduction of a system model leads to controller cost minimization (Garsia, 1998). An example is the ICs, where increasing package density forces developers to include side effects. Knowing that these devices are often modeled by large RLC circuits, this would be too demanding computationally and practically due to the detailed modeling of the original system (Benner, 2007). In control system, due to the fact that feedback controllers do not usually consider all the dynamics of the system, model reduction becomes a very important issue (Bui-Thanh, 2005).

For a reduced model order that will best mimic the performance of its original system, system transformation is performed. In the process of system transformation, some system parameters are required to be identified. This objective maybe achieved by the use of artificial neural networks (ANN) (Alsmadi, 2007), which are considered as the new generation of information processing networks (Hinton, 2006). Artificial neural systems maybe defined as physical cellular systems which have the capability of acquiring, storing and utilizing experiential knowledge. They can be represented as mathematical or computational models based on biological neural networks. An artificial neural network consists of an interconnected group of artificial neurons and processes information. They perform summing operations and nonlinear function computations. Neurons are usually organized in layers and forward connections where computations are performed in a parallel fashion at all nodes and

connections. Each connection is expressed by a numerical value called a weight. The learning process of a neuron corresponds to a way of changing its weights. An artificial neural network can be used to model complex relationships between inputs and outputs of different systems (Haykin, 1994) (Zurada, 1992) (Williams, 1989).

In obtaining the overall transformed model, which leads to control design advantages, part of the transformation requires some optimized solution. This is accomplished using what is called the linear matrix inequality (LMI), which serves application problems, in convex optimization (Boyd, 1994). The LMI is based on the Lyapunov theory of showing that the differential equation $\dot{x}(t) = Ax(t)$ is stable if and only if there exists a positive definite matrix [**P**] such that $A^T P + PA < 0$. The requirement $\{ P > 0,$ $A^T P + PA < 0 \}$ is what is known as Lyapunov inequality on [**P**]. The LMIs that arise in systems and control theory can be formulated as convex optimization problems that are amenable to computer solution and then solved using different algorithms (Boyd, 1994).

This paper is organized as follows: Section 2 presents background on model order reduction and artificial neural networks. A detailed illustration of the ANN transformed system state matrix estimation and the LMI-based complete system transformation is presented in Section 3. Section 4 presents a practical implementation of the ANN transformation training, LMI-based transformation, and singular perturbation reduction along with simulation comparative results. Conclusions are presented in Section 5.

2 PRELIMINARY

Many of linear time-invariant (LTI) systems have fast and slow dynamics, which are referred to as singularly perturbed systems (Kokotovic, Khalil, and O'Reilly, 1986). Neglecting the fast dynamics gives the advantage of designing simpler lower-dimensionality reduced order controllers. To show the formulation of a reduced model order, consider the following system:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1}$$

$$y(t) = Cx(t) + Du(t) \tag{2}$$

As a singularly perturbed system (with slow and fast dynamics), Equations (1) - (2) may be formatted as:

$$\dot{x}(t) = A_{11}x(t) + A_{12}\xi(t) + B_1 u(t), \quad x(0) = x_0 \tag{3}$$

$$\varepsilon\dot{\xi}(t) = A_{21}x(t) + A_{22}\xi(t) + B_2 u(t), \quad \xi(0) = \xi_0 \tag{4}$$

$$\mathrm{y}(t) = C_1 x(t) + C_2 \xi(t) + Du(t) \tag{5}$$

where $x \in \Re^{m_1}$ and $\xi \in \Re^{m_2}$ are the slow and fast state variables respectively, $u \in \Re^{n_1}$ and $y \in \Re^{n_2}$ are the input and output vectors respectively, $\{[\mathbf{A_{ii}}], [\mathbf{B_i}], [\mathbf{C_i}], [\mathbf{D}]\}$ are constant matrices of appropriate dimensions with $i \in \{1, 2\}$, and ε is a small positive constant. The singularly perturbed system in Equations (3)-(5) is simplified by setting $\varepsilon = 0$. That is, the fast dynamics of the system are being neglected and the state variables ξ are assumed to have reached their quasi-steady state. Hence, setting $\varepsilon = 0$ in Equation (4), with the assumption that $[\mathbf{A_{22}}]$ is nonsingular, produces:

$$\xi(t) = -A_{22}^{-1}A_{21}x_r(t) - A_{22}^{-1}B_1 u(t) \tag{6}$$

where the index *r* denotes the remained or reduced model. Substituting Equation (6) in Equations (3)-(5) yields the following reduced model order:

$$\dot{x}_r(t) = A_r x_r(t) + B_r u(t) \tag{7}$$

$$y(t) = C_r x_r(t) + D_r u(t) \tag{8}$$

Where the new matrices: $A_r = A_{11} - A_{12}A_{22}^{-1}A_{21}$, $B_r = B_1 - A_{12}A_{22}^{-1}B_2$, $C_r = C_1 - C_2 A_{22}^{-1}A_{21}$, and $D_r = D - C_2 A_{22}^{-1}B_2$.

The system in Equations (1) and (2) maybe estimated by an ANN. In this paper, a recurrent neural network based on an approximation of the method of steepest descent is used for the estimation of the system state matrix. The network tries to match the output of certain neurons with the desired values of the system output at specific instant of time (Haykin, 1994) (Williams, 1989). Hence, consider the discrete system given by:

$$x(k+1) = A_d x(k) + B_d u(k) \tag{9}$$

$$y(k) = x(k) \tag{10}$$

which, for a system with two eigenvalue categories (slow and fast), can be represented as:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix} u(k) \tag{11}$$

$$y(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \tag{12}$$

where k is the time index. Using the recurrent neural network, the system in Equations (11) and (12) for a 2nd model order can be estimated as illustrated in Figure 1.

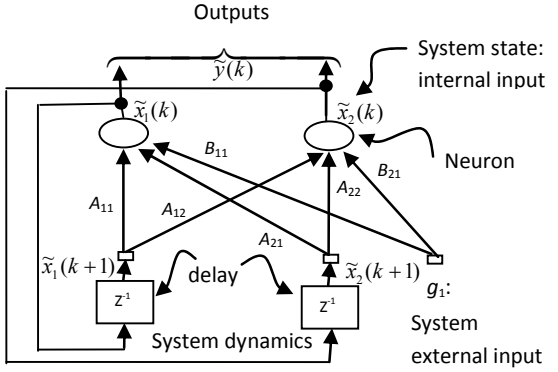


Figure 1: A second order recurrent neural network architecture.

As a general case, consider a network consisting of a total of N neurons with M external input connections, as shown in Figure 1 for a 2nd model order. Let the variable $\mathbf{g}(k)$ denotes the (M x 1) external input vector applied to the network at discrete time k and the variable $\mathbf{y}(k + 1)$ denotes the corresponding (N x 1) vector of individual neuron outputs produced one step later at time (k + 1). The input vector $\mathbf{g}(k)$ and one-step delayed output vector $\mathbf{y}(k)$ are concatenated to form the ((M + N) x 1) vector $\mathbf{u}(k)$, whose i^{th} element is denoted by $u_i(k)$. If Λ denotes the set of indices i for which $g_i(k)$ is an external input, and β denotes the set of indices i for which $u_i(k)$ is the output of a neuron (which is $y_i(k)$), the following is true:

$$u_i(k) = \begin{cases} g_i(k), & \text{if } i \in \Lambda \\ y_i(k), & \text{if } i \in \beta \end{cases} \tag{13}$$

The (N x (M + N)) recurrent weight matrix of the network is represented by the variable [**W**]. The net internal activity of neuron j at time k is given by:

$$v_j(k) = \sum_{i \in \Lambda \cup \beta} w_{ji}(k) u_i(k) \tag{14}$$

At (k + 1), the output of the neuron j is computed by passing $v_j(k)$ through the nonlinearity $\varphi(.)$:

$$y_j(k+1) = \varphi(v_j(k)) \tag{15}$$

The derivation of the recurrent algorithm maybe obtained by using $d_j(k)$ to denote the desired response of neuron j at time k, and $\varsigma(k)$ to denote the set of neurons that are chosen to provide externally reachable outputs. A time-varying (N x 1) error vector $\boldsymbol{e}(k)$ is defined whose j^{th} element is given by the following relationship:

$$e_j(k) = \begin{cases} d_j(k) - y_j(k), & \text{if } j \in \varsigma(k) \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

The objective is to minimize the cost function E_{total} which is obtained by:

$$E_{\text{total}} = \sum_k E(k) = \sum_k [\frac{1}{2} \sum_{j \in \varsigma} e_j^2(k)] \tag{17}$$

This cost function will be minimized by estimating the instantaneous gradient, which is the error at each instant of time k with respect to the weight matrix [**W**] and then updating [**W**] in the negative direction of this gradient (Haykin, 1994). As a result:

$$W = \left[[\tilde{A}_d] \; [\tilde{B}_d] \right] \tag{18}$$

where $\tilde{\mathrm{A}}_{\mathrm{d}}$ and $\tilde{\mathrm{B}}_{\mathrm{d}}$ are the estimates of Equation (9).

3 SYSTEM TRANSFORMATION AND ORDER REDUCTION

In the new reduction technique, the system is transformed before the model order is reduced. System transformation is achieved by transforming the system state matrix [**A**] based on the ANN estimation and then transforming the [**B**], [**C**], and [**D**] matrices of Equations (1) and (2) using the LMI-based transformation.

3.1 ANN System State Matrix Transformation

In this paper, one objective is to search for a transformation that decouples different categories of system eigenvalues. In the transformed system presented in this paper, the dominant eigenvalue category is selected as a subset of the original system eigenvalues. This is accomplished by transforming the system state matrix [**A**] in Equation (1) into [$\hat{\mathrm{A}}$] (for all real eigenvalues) as follows:

$$\hat{A} = \begin{bmatrix} \lambda_1 & a_{12} & \cdots & a_{1n} \\ 0 & \lambda_2 & \cdots & a_{2n} \\ \vdots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix} \quad (19)$$

This is an upper triangular matrix that has the original system eigenvalues preserved in the diagonal, seen as λ_i, and has the elements to be identified, seen as (a_{ij}). It is set as such for the purpose of eliminating the fast dynamics and sustaining the slow dynamics through model order reduction. In order to evaluate the (a_{ij}) elements, first, the system of Equations (1) and (2) is discretized as shown in Equations (9) and (10), second, the [A_d] in Equation (9) is transformed into [$\widetilde{A}_d$] (similar to the form seen in Equation (19)), third, the recurrent neural network estimates the required elements of [$\widetilde{A}_d$], fourth, (a_{ij}) are then evaluated once the continuous form is obtained from the estimated discrete system.

In this estimation, the interest is to estimate or obtain the [$\widetilde{A}_d$] only without the estimation of the [$\widetilde{B}_d$] matrix, where this [$\widetilde{B}_d$] matrix is automatically obtained in the recurrent network as seen in Figure 1 and Equation (18). In order to achieve this objective, the zero input ($u(k)$ = 0) response is obtained where the input/output data is basically generated based on the initial state conditions only. Hence, the discrete system of Equations (9) and (10), with initial state conditions $x(0) = x_0$, becomes:

$$x(k+1) = A_d x(k), \qquad x(0) = x_0 \quad (20)$$

$$y(k) = x(k) \quad (21)$$

Now based on Equations (20) and (21), where the initial states are the system input and the obtained states are the system output, a set of input/output data is obtained and the neural network estimation is applied (Haykin, 1994). In steps:

Step 1. Initialize the weights [**W**] by a set of uniformly distributed random numbers. Starting at the instant k = 0, use Equations (14) and (15) to compute the output values of the N neurons (where $N = \beta$).

Step 2. For every time step k and all $j \in \beta$, $m \in \beta$ and $\ell \in \beta \cup \Lambda$, compute the system dynamics which are governed by the triply indexed set of variables:

$$\pi^j_{m\ell}(k+1) = \dot{\varphi}(v_j(k)) \left[\sum_{i \in \beta} w_{ji}(k) \pi^i_{m\ell}(k) + \delta_{mj} u_\ell(k) \right] \quad (22)$$

with initial conditions $\pi^j_{m\ell}(0) = 0$ and δ_{mj} given by $\left(\partial w_{ji}(k) / \partial w_{m\ell}(k)\right)$ is equal to "1" only when $j = m$ and $i = \ell$; otherwise it is "0". Notice that for the special case of a sigmoidal nonlinearity in the form of a logistic function, the derivative $\dot{\varphi}(\bullet)$ is given by $\dot{\varphi}(v_j(k)) = y_j(k+1)[1 - y_j(k+1)]$.

Step 3. Compute the weight changes corresponding to the error signal and system dynamics:

$$\Delta w_{m\ell}(k) = \eta \sum_{j \in \varsigma} e_j(k) \pi^j_{m\ell}(k) \quad (23)$$

Step 4. Update the weights in accordance with:

$$w_{m\ell}(k+1) = w_{m\ell}(k) + \Delta w_{m\ell}(k) \quad (24)$$

Step 5. Repeat the above 4 steps for final desired estimation.

Training the network as illustrated, produces the discrete transformed system state matrix [$\widetilde{A}_d$]. This new discrete matrix is then converted to the continuous form to give the transformed system state matrix [$\hat{A}$] as actually seen in Equation (19).

3.2 LMI-based Complete System Transformation

The transformation in Equation (19) is motivated by the matrix reducibility concept illustrated as follows (Boyd, 1994) (Horn, 1985):

Definition. A matrix $A \in M_n$ is called reducible if either:

(a) n = 1 and A = 0; or

(b) $n \geq 2$, there is a permutation matrix $P \in M_n$, and some integer r with $1 \leq r \leq n-1$ such that:

$$P^{-1} A P = \begin{bmatrix} X & Y \\ \mathbf{0} & Z \end{bmatrix} \quad (25)$$

where $X \in M_{r,r}$, $Z \in M_{n-r,n-r}$, $Y \in M_{r,n-r}$, and $\mathbf{0} \in M_{n-r,r}$ is a zero matrix.

The attractive features of the permutation matrix [**P**] such as being orthogonal and invertible have made this transformation easy to carry out. Based on the LMI technique, the optimization problem is casted as follows:

$$\min_{P} \left\| P - P_o \right\| \text{ subject to } \left\| P^{-1}AP - \hat{A} \right\| < \varepsilon \tag{26}$$

which maybe written in an LMI equivalent form as:

$$\begin{aligned} &\min_{S} trace(S) \text{ subject to} \\ &\begin{bmatrix} S & P - P_o \\ (P - P_o)^T & I \end{bmatrix} > 0 \\ &\begin{bmatrix} \varepsilon_1^2 I & P^{-1}AP - \hat{A} \\ (P^{-1}AP - \hat{A})^T & I \end{bmatrix} > 0 \end{aligned} \tag{27}$$

where S is a symmetric slack matrix (Boyd, 1994).

The Linear Matrix Inequalities (LMI) are applied to the [**A**] and [$\hat{\text{A}}$] matrices and the transformation matrix [**P**] is then obtained, which is necessary for obtaining the complete system transformation {[$\hat{\mathbf{B}}$], [$\hat{\mathbf{C}}$], [$\hat{\mathbf{D}}$]}. Complete system transformation can be achieved as follows: assuming that $\hat{x} = P^{-1}x$, the system of Equations (1) and (2) can be re-written as:

$$P\dot{\hat{x}}(t) = AP\hat{x}(t) + Bu(t) \tag{28}$$

$$\hat{y}(t) = CP\hat{x}(t) + Du(t) \tag{29}$$

Pre-multiplying Equation (28) by [$\mathbf{P^{-1}}$] yields:

$$\begin{aligned} &P^{-1}P\dot{\hat{x}}(t) = P^{-1}AP\hat{x}(t) + P^{-1}Bu(t) \\ &\therefore\ \dot{\hat{x}}(t) = \hat{A}\hat{x}(t) + \hat{B}u(t) \end{aligned} \tag{30}$$

$$\begin{aligned} \text{and} \qquad &\hat{y}(t) = CP\hat{x}(t) + Du(t) \\ &\therefore\ \hat{y}(t) = \hat{C}\hat{x}(t) + \hat{D}u(t) \end{aligned} \tag{31}$$

where the transformed system matrices are:
$\hat{A} = P^{-1}AP$, $\hat{B} = P^{-1}B$, $\hat{C} = CP$, and $\hat{D} = D$.

3.3 Model Order Reduction

Model order reduction will now be applied to the system of Equations (30) and (31) which has the following format:

$$\begin{bmatrix} \dot{\hat{x}}_r(t) \\ \dot{\hat{x}}_o(t) \end{bmatrix} = \begin{bmatrix} A_{rn} & A_c \\ 0 & A_o \end{bmatrix} \begin{bmatrix} \hat{x}_r(t) \\ \hat{x}_o(t) \end{bmatrix} + \begin{bmatrix} B_r \\ B_o \end{bmatrix} u(t) \tag{32}$$

$$\hat{y}(t) = \begin{bmatrix} C_{rn} & C_o \end{bmatrix} \begin{bmatrix} \hat{x}_r(t) \\ \hat{x}_o(t) \end{bmatrix} + \hat{D}u(t) \tag{33}$$

Notice that in the new formulation, the dominant eigenvalues (slow dynamics) which are presented in A_{rn} are now decoupled from the non-dominant eigenvalues (fast dynamics) which are presented in A_o. Hence, as illustrated in Equations (3) and (4) for order reduction, Equation (32) is written as:

$$\dot{\hat{x}}_r(t) = A_{rn}\hat{x}_r(t) + A_c\hat{x}_o(t) + B_r u(t) \tag{34}$$

$$\dot{\hat{x}}_o(t) = A_o\hat{x}_o(t) + B_o u(t) \tag{35}$$

By neglecting the system fast dynamics (setting $\dot{\hat{x}}_o(t) = 0$ by setting $\varepsilon = 0$)), the coupling term $A_c\hat{x}_o(t)$ is evaluated by solving for $\hat{x}_o(t)$ in Equation (35). That is, $\hat{x}_o(t) = -A_o^{-1}B_o u(t)$ and the reduced model order becomes:

$$\dot{\hat{x}}_r(t) = A_{rn}\hat{x}_r(t) + [-A_c A_o^{-1}B_o + B_r]u(t) \tag{36}$$

$$\hat{y}(t) = C_r\hat{x}_r(t) + [-C_o A_o^{-1}B_o + D]u(t) \tag{37}$$

Hence, the overall transformed reduced model order is given by:

$$\dot{\hat{x}}_r(t) = A_{or}\hat{x}_r(t) + B_{or}u(t) \tag{38}$$

$$\hat{y}(t) = C_{or}\hat{x}_r(t) + D_{or}u(t) \tag{39}$$

where the details of the {[$\mathbf{A_{or}}$], [$\mathbf{B_{or}}$], [$\mathbf{C_{or}}$], [$\mathbf{D_{or}}$]} overall reduced matrices are shown in Equations (36) and (37).

4 SIMULATIONS AND RESULTS

The proposed method of reduced order system modeling based on neural network estimation, LMI-based transformation, and model order reduction is investigated the following case studies.

Case Study. Consider the system of a high-performance tape transport shown in Figure 2 (Franklin, 1994). The system is designed with a small capstan to pull the tape past the read/write heads with the take-up reels turned by DC motors. In the static equilibrium, the tape tension equals the vacuum force $T_o = F$ and the torque from the motor equals the torque on the capstan $K_t i_o = r_1 T_o$. Please notice that all the variables are defined in (Franklin, 1994).

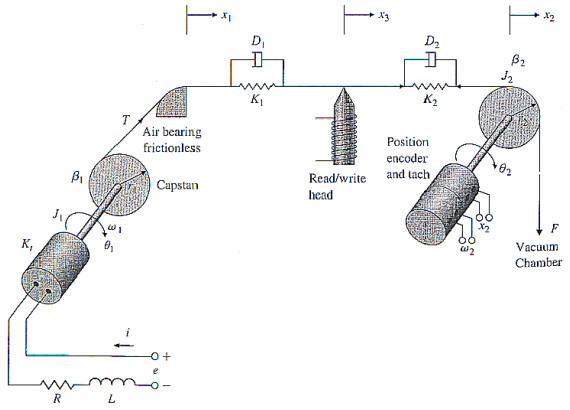


Figure 2: Tape-drive system schematic control model.

The variables are defined as deviations from the equilibrium. The system equations of motion are given as follows:

$$J_1 = \frac{d\omega_1}{dt} + \beta_1\omega_1 - r_1 T + K_t i\,,\ \dot{x}_1 = r_1\omega_1$$

$$L\frac{di}{dt} Ri + K_e\omega_1 = e\,,\ \ \dot{x}_2 = r_2\omega_2$$

$$J_2\frac{d\omega_2}{dt} + \beta_2\omega_2 + r_2 T = 0$$

$$T = K_1(x_3 - x_1) + D_1(\dot{x}_3 - \dot{x}_1)$$

$$T = K_2(x_2 - x_3) + D_2(\dot{x}_2 - \dot{x}_3)$$

$$x_1 = r_1\theta_1\,,\ \ x_2 = r_2\theta_2\,,\ \ x_3 = \frac{x_1 - x_2}{2}\,,$$

The state space model is derived from the system equations, where there are (i) one input, which is the applied voltage, (ii) three outputs, which are: (1) tape position at the head, (2) tape tension, and (3) tape position at the wheel, (iii) five states: (1) tape position at the air bearing, (2) drive wheel speed, (3) tape position at the wheel, (4) tachometer output speed, and (5) capstan motor speed. For dynamical testing of the new reduction technique validity, different cases of this practical system were investigated.

As a first example, a system with all real eigenvalues is considered:

$$\dot{x}(t) = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 \\ -0.1 & -1.35 & 0.1 & 04.1 & 0.75 \\ 0 & 0 & 0 & 5 & 0 \\ 0.35 & 0.4 & -1.4 & -5.4 & 0 \\ 0 & -0.03 & 0 & 0 & -10 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 & 0 \\ -0.2 & -0.2 & 0.2 & 0.2 & 0 \end{bmatrix} x(t)$$

with the eigenvalues {-9.9973, -3.9702, -1.8992, -0.677, -0.2055}. Since there are two categories of eigenvalues, slow {-1.8992, -0.6778, -0.2055} and fast {-9.9973, -3.9702}, model order reduction may be applied.

Discretizing this system with a sampling period T_s = 0.1s, simulating the discrete system for 200 input/output data points, and training it with learning rate of $\eta = 1 \times 10^{-4}$ and initial weights for $[\tilde{\mathbf{A}}_\mathbf{d}]$:

$$w = \begin{bmatrix} 0.0048 & 0.0039 & 0.0009 & 0.0089 & 0.0168 \\ 0.0072 & 0.0024 & 0.0048 & 0.0017 & 0.0040 \\ 0.0176 & 0.0176 & 0.0136 & 0.0175 & 0.0034 \\ 0.0055 & 0.0039 & 0.0078 & 0.0076 & 0.0051 \\ 0.0102 & 0.0024 & 0.0091 & 0.0049 & 0.0121 \end{bmatrix}$$

produces the transformed system matrix:

$$\hat{A} = \begin{bmatrix} -0.2051 & -0.0367 & -0.0068 & 0.0762 & 0.2074 \\ 0 & -0.6782 & 0.0513 & 0.0156 & 0.0554 \\ 0 & 0 & -1.8986 & 0.2282 & 0.0537 \\ 0 & 0 & 0 & -3.9708 & 0.0920 \\ 0 & 0 & 0 & 0 & -9.9963 \end{bmatrix}$$

with estimated eigenvalues -9.9963, -3.9708, -1.898, -0.6782, -0.2051. This was achieved by decoupling the fast eigenvalue category from the slow one, which simply was done by first placing the slow eigenvalue category in λ_i of Equation (19) and then the fast category. As observed in $\hat{A}$ above, the eigenvalues are almost identical with the original system with little difference due to discretization. Using the LMI-based system transformation, the complete transformed system is obtained. Considering the {-9.9963, -3.9708} as the fast category eigenvalue, the 3rd order reduced model is determined. Simulation results based on (i) model order reduction without system transformation, (ii) model order reduction with ANN transformation (estimation of $[\tilde{\mathbf{A}}_\mathbf{d}]$ and $[\tilde{\mathbf{B}}_\mathbf{d}]$ matrices only as presented in Equations (11) and (12)), (iii) model order reduction with LMI-based complete system transformation, and (iv) the original 5th order system are all shown in Figure 3.

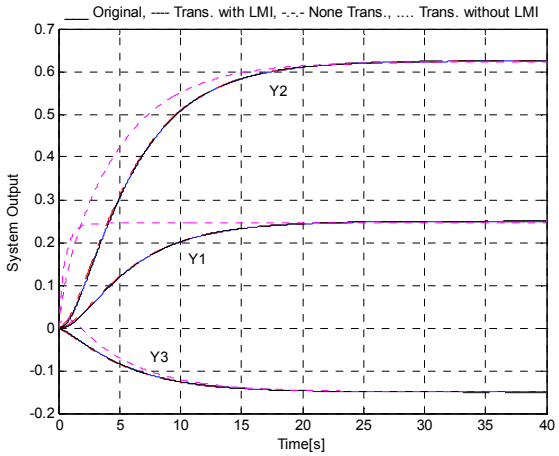


Figure 3: Reduced 3rd model orders (Pink….. transformed with ANN estimation only, Red-.-.-.- non-transformed, Black---- transformed with LMI) output responses to a step input along with the non reduced (Blue____ original) system output response. The LMI-transformed curve fits almost exactly on the original response.

For more rigorous testing of the new reduction technique, the 5th model order is reduced to a 2nd order assuming that the -1.8986 belongs to the fast eigenvalue category. Hence, the 2nd order reduced model with its eigenvalues preserved as desired is obtained:

$$\dot{\hat{x}}_r(t) = \begin{bmatrix} -0.2051 & -0.0367 \\ 0 & -0.6782 \end{bmatrix} \hat{x}_r(t) + \begin{bmatrix} -1.9672 \\ -2.1764 \end{bmatrix} u(t)$$

$$\hat{y}(t) = \begin{bmatrix} -0.0436 & 0.0451 \\ -0.1055 & 0.1029 \\ 0.0217 & -0.0140 \end{bmatrix} \hat{x}_r(t) + \begin{bmatrix} 0.0018 \\ 0.0043 \\ 0.0005 \end{bmatrix} u(t)$$

Simulating this reduced 2nd model order as performed for the 3rd model order, provided the results shown in Figure 4 where the new reduction technique results in responses are identical to the original system's.

As a second example, the system considered here consists of two complex eigenvalues and three real,

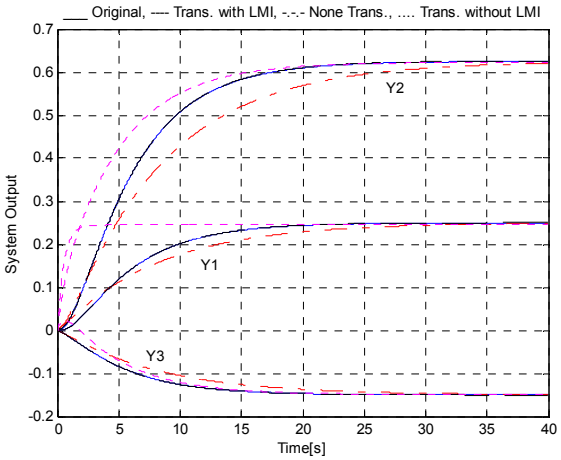


Figure 4: Plots of Pink…. 3rd order transformed with ANN estimation only and reduced 2nd model orders (Red-.-.-.- non-transformed, Black---- transformed with LMI) output responses to a step input along with the non reduced (Blue____ original) system output response. The LMI-transformed curve fits almost exactly on the original response.

where two of the real eigenvalues produce fast dynamics. The system is given by:

$$\dot{x}(t) = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 \\ -1.1 & -1.35 & 1.1 & 3.1 & 0.75 \\ 0 & 0 & 0 & 5 & 0 \\ 1.35 & 1.4 & -2.4 & -11.4 & 0 \\ 0 & -0.03 & 0 & 0 & -10 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(t)\,,$$

$$y(t) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 & 0 \\ -0.2 & -0.2 & 0.2 & 0.2 & 0 \end{bmatrix} x(t)$$

The five eigenvalues are {-10.5772, -9.999, -0.9814, -0.5962 ± j0.8702}. Considering the {-10.5772, -9.999} as the fast eigenvalue category, model order reduction is performed.

Discretizing the system with T_s = 0.1s, using a step input with a learning time T_l = 15s, and training the ANN for the input/output data with η = 0.001 learning rate produces the transformed system matrix:

$$\hat{A} = \begin{bmatrix} -0.5967 & 0.8701 & -1.4633 & -0.9860 & 0.0964 \\ -0.8701 & -0.5967 & 0.2276 & 0.6165 & 0.2114 \\ 0 & 0 & -0.9809 & 0.1395 & 0.4934 \\ 0 & 0 & 0 & -9.9985 & 1.0449 \\ 0 & 0 & 0 & 0 & -10.5764 \end{bmatrix}$$

As observed, all the system eigenvalues have been preserved. Based on this transformed matrix, using the LMI technique, the permutation matrix [**P**] is computed and then used for obtaining the [$\hat{\mathbf{B}}$], [$\hat{\mathbf{C}}$], and [$\hat{\mathbf{D}}$] matrices. Since there are two eigenvalues that produce fast dynamics, the following 3rd order reduced model is obtained:

$$\dot{\hat{x}}_r(t) = \begin{bmatrix} -0.5967 & 0.8701 & -1.4633 \\ -0.8701 & -0.5967 & 0.2276 \\ 0 & 0 & -0.9809 \end{bmatrix} \hat{x}_r(t) + \begin{bmatrix} 35.1670 \\ -47.3374 \\ -4.1652 \end{bmatrix} u(t)$$

$$\hat{y}(t) = \begin{bmatrix} -0.0019 & 0 & -0.0139 \\ -0.0024 & -0.0009 & -0.0088 \\ -0.0001 & 0.0004 & -0.0021 \end{bmatrix} \hat{x}_r(t) + \begin{bmatrix} -0.0025 \\ -0.0025 \\ 0.0006 \end{bmatrix} u(t)$$

The reduced model has also preserved the original system dominant eigenvalues {-0.9809, -0.5967± j0.8701}, which achieves the proposed objective. Investigating the performance of this reduced model order compared with the other reduction techniques shows again its superiority as seen in Figure 5. The LMI-based transformed responses are almost identical to the 5th order original systems'.

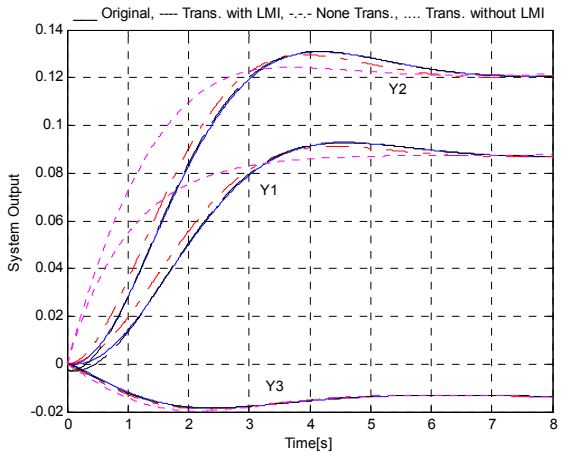


Figure 5: Reduced 3rd model orders (Pink…. transformed with ANN estimation only, Red-.-.-.- non-transformed, Black---- complete transformation with LMI) output responses to a step input along with the non reduced (Blue____ original) system output response. The LMI-transformed curve fits almost exactly on the original response.

5 CONCLUSIONS

In this paper, a new method of dynamic systems model order reduction is presented that has the following advantages. First, in the transformed model, a decoupling of the slow and fast dynamics is achieved. Second, in the reduced model order, the eigenvalues are preserved as a subset of the original system. Third, the reduced model order shows responses that are usually almost identical to the original full order system. Hence, observing the simulation results, it is clear that modeling of dynamic systems using the new LMI-based reduction technique is superior to those other reduction techniques.

REFERENCES

Alsmadi, O., Abdalla, M., 2007. Model order Reduction for Two-Time-Scale Systems Based on Neural Network Estimation. *Proceedings of the 15th Mediterranean Conference on Control & Automation*. Athens, Greece.

Benner, P., 2007. Model Reduction at ICIAM'07. *SIAM News*, V. 40, N. 8.

Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V., 1994. *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial and Applied Mathematics (SIAM).

Bui-Thanh, T., Willcox, K., 2005. Model Reduction for Large-Scale CFD Applications using the Balanced Proper Orthogonal Decomposition. *17th American Institute of Aeronautics and Astronautics Computational Fluid Dynamics Conference*. Toronto, Canada.

Franklin, G., Powell, J., Emami-Naeini, A., 1994. *Feedback Control of Dynamic Systems*, Addison-Wesley, 3rd edition.

Garsia, G., Dfouz, J., Benussou, J., 1998. H2 Guaranteed Cost Control for Singularly Perturbed Uncertain Systems. *IEEE Transactions on Automatic Control*, Vol. 43, pp. 1323-1329.

Haykin, S., 1994. *Neural Networks*: *a Comprehensive Foundation*, Macmillan College Publishing Company, New York.

Hinton, G., Salakhutdinov, R., 2006. Reducing the Dimensionality of Data with Neural Networks. *Science*, pp. 504-507.

Horn, R., Johnson, C., 1985. *Matrix Analysis*. Cambridge, UK.

Kokotovic, P., Khalil, H., O'Reilly, J., 1986. *Singular Perturbation Methods in Control: Analysis and Design*, Academic Press. Orlando, Florida.

Williams, R., Zipser, D., 1989. A Learning Algorithm for Continually Running Full Recurrent Neural Networks. *Neural Computation,* 1(2), pp. 270-280.

Zhou, W., Shao, S., Gao, Z., 2009. A Stability Study of the Active Disturbance Rejection Control Problem by a Singular Perturbation Approach. *Applied Mathematical Science*, Vol. 3, no. 10, pp. 491-508.

Zurada, J., 1992. *Artificial Neural Systems,* West Publishing Company, New York.

THEORETICAL CALCULATION OF THERMAL CONTACT RESISTANCE OF BALL BEARING UNDER DIFFERENT LOADS

Chao Jin, Bo Wu
State Key Laboratory for Digital Manufacturing Equipment and Technology
Huazhong University of Science and Technology, Wuhan, Hubei 430074, China

Youmin Hu
School of Mechanical Science and Engineering, Huazhong University of Science and Technology
Wuhan, Hubei 430074, China

Keywords: Hertzian theory, JHM method, Load distribution, Thermal contact resistance.

Abstract: The thermal contact resistance between the balls and the inner and outer rings of an angular contact ball bearing is investigated. It is assumed that the bearing sustains thrust, radial, or combined loads under a steady-state temperature condition. The shapes and sizes of the contact areas are calculated using the Hertzian theory. The distribution of internal loading in the bearing is determined by the JHM method. The comparison between the experimental data and the calculated values confirms the validity of the prediction method for the thermal contact resistances between the elements of a bearing.

1 INTRODUCTION

In a high-speed feeding system, bearings are considered to be the main heat sources, and the thermal properties of the bearings need to be carefully studied. For a bearing, the thermal resistances for conduction through the bearing elements themselves and for radiation can be calculated using the dimensions, the thermal conductivities, the thermal-optical properties, and the temperatures of the elements. However, it ca be said that the thermal contact resistances between the balls and the rings, which are most closely related to the temperature differences across the bearings, are difficult to predict because few useful calculation method have been proposed yet.

Since the thermal resistance results from the fact that most of the heat is constrained to flow through small contact areas, a reasonable step in determining the contact resistance between the balls and the inner and outer rings of the bearing would be to use a similar approach to that adopted to solve the thermal constriction problem for ideal smooth surfaces. The thermal constriction resistances for circular, circular annular, rectangular, and other geometrical-shaped contact areas are normally solved analytically or numerically as Dirichlet problems. The prediction of the thermal contact resistance necessitates the determination of the contact area. This is possible with the Hertzian theory when the contact surfaces are approximated as being smooth. In addition to the study by Clausing and Chao, the thermal contact resistance problem has been discussed in many papers. Most papers determine the contact areas using the Hertzian theory. However, a survey of the literature shows that only the studies by Yovanovich have dealt with the problem of the contact resistance between bearing elements. He studied the contact resistance under axial loads and concluded that the contact resistance depends on the size and shape of contact area as determined by the Hertzian theory and the thermal conductivity of the material. He did not, however, give thermal designers a tractable expression that considered the change in contact angle induced by elastic deformation at the contact points. Also, he did not consider other types of loadings such as radial and combined axial/radial loads.

This article develops an approach that accurately predicts the thermal contact resistance between the balls and the inner and outer rings of an angular contact ball bearing. The contact forces required to calculate the contact area are explicitly formulated for axial, radial, and combined loadings. The prediction method for the thermal contact resistance is verified by comparing the calculated values with

experimental results measured in a high-speed feeding system.

2 EXPRESSIONS FOR CONTACT RESISTANCE

2.1 Contact Resistance

The contact resistances between the balls and the inner and outer rings may be treated in the same manner as constriction resistance since both resistances result from the restriction of the heat flow due to small contact arrears. Thus, the assumptions utilized to solve the constriction resistance may be applicable to the present problem. It is assumed that one half of the thermal constriction resistance problem can be adequately represented by an isolated, isothermal area either supplying or receiving heat from an other-wise insulated conducting half-space. In the ellipsoidal coordinate system the Laplace's equation is:

$$\nabla^2 T = \frac{\partial}{\partial u}(\sqrt{f(u)}\frac{\partial T}{\partial u}) \tag{1}$$

Where

$$\sqrt{f(u)} = \sqrt{(a^2+u)(b^2+u)u} \tag{2}$$

And a, b are the semi-major and semi-minor axes of the elliptic contact area, respectively; while u is the variable along an axis normal to the contact plane. The boundary conditions are:

$$u = 0, T = T_0 \text{ , const} \tag{3}$$

$$u \to \infty, T = 0 \tag{4}$$

With Equation (1), (3) and (4), the temperature distribution can be obtained:

$$T = \frac{Q}{4\pi k}\int_u^\infty \frac{du}{\sqrt{f(u)}} \tag{5}$$

Where Q is all the heat leaving the elliptic contact area, and by the definition of the thermal contact resistance:

$$R = \frac{T_0 - T_{u\to\infty}}{Q} = \frac{1}{4\pi k}\int_0^\infty \frac{du}{\sqrt{f(u)}} \tag{6}$$

Using the complete elliptic integral of the first kind, Equation (6) can be written in the following form as:

$$R = \frac{\Psi(a/b)}{4ka}, \Psi(a/b) = \frac{2}{\pi}F(e,\frac{\pi}{2}) \tag{7}$$

Then, the contact thermal resistance between the ball and the inner or outer ring can be determined by using Equation (7). For most bearing, whose ball and both rings are made from the same material, i.e., $k = k_b = k_i = k_o$, we can write the contact thermal resistance per ball as:

$$R = \frac{1}{2k}[\frac{\Psi(a_i/b_i)}{a_i} + \frac{\Psi(a_o/b_o)}{a_o}] \tag{8}$$

These expressions permit us to predict the total contact resistance resulting from the contact of an arbitrary number of balls with both the inner and outer rings by connecting the thermal resistances in parallel.

2.2 Contact Areas in a Ball Bearing

The thermal contact resistance is generally considered as a function of the shape and size 1of the contact area. When two elastic bodies having smooth round surface are press against each other, the contact area becomes elliptic. The formulations that determine the semi-major and semi-minor axes of the elliptic contact area are summarized herein. In deriving the following eaxpressions, it is assumed that the angle between the two planes containing the principal radii of curvature of the bodies are perpendicular as in the case of balls contacting the inner or outer ring of a bearing:

$$a = a^*[\frac{3}{4}\frac{P}{A+B}(\frac{1-\nu_1^2}{E_1} + \frac{1-\nu_2^2}{E_2})]^{1/3}$$

$$b = b^*[\frac{3}{4}\frac{P}{A+B}(\frac{1-\nu_1^2}{E_1} + \frac{1-\nu_2^2}{E_2})]^{1/3} \tag{9}$$

$$A = \frac{1}{2}(\frac{1}{r_1} + \frac{1}{r_2}), B = \frac{1}{2}(\frac{1}{r_1'} + \frac{1}{r_2'}) \tag{10}$$

In which r_1, r_1' are the radius of curvature for inner or outer race and groove, respectively. And r_2, r_2' are the radii of rolling ball. Considering the bearing model shown in Figure1, for the contact at inner ring side, the radius of curvature r_1' of the inner groove must be treated as negative in Equation (10); while at the outer ring side contact, r_1, r_1' must be treated as negative.

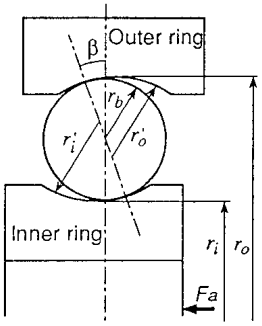


Figure 1: Schematic of bearing.

The values of a^* and b^* are calculated as follows:

$$\begin{cases} I = \frac{2}{e^2}[F(e,\frac{\pi}{2}) - E(e,\frac{\pi}{2})] \\ J = \frac{2}{e^2}[\frac{E(e,\frac{\pi}{2})}{1-e^2} - F(e,\frac{\pi}{2})] \\ \frac{J}{I} = \frac{A}{B} \end{cases} \tag{11}$$

In which $F(e,\frac{\pi}{2})$ and $E(e,\frac{\pi}{2})$ are the complete elliptic integrals of the first and second, respectively.

$$\begin{aligned} F(e,\frac{\pi}{2}) &= \int_0^{\pi/2} (1-e^2\sin^2\Phi)^{\frac{-1}{2}} d\Phi \\ E(e,\frac{\pi}{2}) &= \int_0^{\pi/2} (1-e^2\sin^2\Phi)^{\frac{1}{2}} d\Phi \end{aligned} \tag{12}$$

Equation (11) can be solved numerically by the Newton-Downhill method, and then e can be determined, the value of I and J can be calculated. Finally,

$$a^* = (\frac{I+J}{\pi})^{1/3}, b^* = a^*(1-e^2)^{1/2} \tag{13}$$

3 LOAD TYPES AND CONTACT FORCE

We can now use Equation (8) for the prediction of the thermal contact resistance if the contact force for each ball is determined from the total load on the bearing.

3.1 Contact Force Under Centric Thrust Load

Angular contact ball bearings subjected to a centric thrust load have the load distributed equally among the rolling elements. Hence

$$Q = \frac{F_a}{Z\sin\alpha} \tag{14}$$

Where a is the contact angle that occurs in the loaded bearings, and can be determined as follows. In the unloaded condition, the initial contact angle is defined by

$$\cos\alpha^o = 1 - \frac{P_d}{2BD} \tag{15}$$

In which B is the total curvature, and P_d is the mounted diametral clearance.

$$P_d = d_o - d_i - 2D \tag{16}$$

A thrust load F_a applied to the inner ring as shown in Figure2 causes an axial deflection δ_a. This axial deflection is a component of a normal deflection along the line of contact such that from Figure2.

$$\delta_n = BD(\frac{\cos\alpha^o}{\cos\alpha} - 1) \tag{17}$$

Since $Q=K\delta_n^{1.5}$, where K is the load-deflection factor. Substituting Equation (17) into (14), we get,

$$\frac{F_a}{ZK(BD)^{1.5}} = \sin\alpha(\frac{\cos\alpha^o}{\cos\alpha} - 1)^{1.5} \tag{18}$$

Equation (18) may be solved numerically by the Newton-Raphson method, the equation to be satisfied iteratively is,

$$\alpha' = \alpha + \frac{\frac{F_a}{ZK(BD)^{1.5}} - \sin\alpha(\frac{\cos\alpha^o}{\cos\alpha} - 1)^{1.5}}{\cos\alpha(\frac{\cos\alpha^o}{\cos\alpha} - 1)^{1.5} + 1.5\tan^2\alpha\cos\alpha^o(\frac{\cos\alpha^o}{\cos\alpha} - 1)^{0.5}} \tag{19}$$

Equation (19) is satisfied when a'–a is essentially zero. Simultaneously, from Fig.2, we can get

$$\delta_a = \frac{BD\sin(\alpha - \alpha^o)}{\cos\alpha} \tag{20}$$

3.2 Contact Force Under Combined Radial and Thrust Load

If rolling bearing without diametral clearance is subjected simultaneously to a radial load in the central plane of the roller and a centric thrust load, then the inner rings of the bearing will remain parallel and will be relatively displaced a distance δ_a in the axial direction and δ_r in the radial direction. At any position Ψ measured from the most

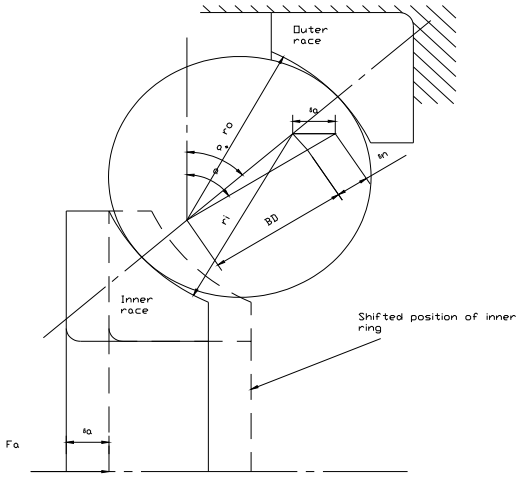


Figure 2: Angular contact ball bearing under thrust load.

heavily loaded rolling element, the approach of the rings is,

$$\delta_\psi = \delta_a \sin\alpha + \delta_r \cos\alpha \cos\psi \tag{21}$$

At Ψ=0 maximum deflection occurs and is given by

$$\delta_{\max} = \delta_a \sin\alpha + \delta_r \cos\alpha \tag{22}$$

Combining Equation (21) and (22) yields

$$\delta_\psi = \delta_{a\max}[1 - \frac{1}{2\varepsilon}(1-\cos\psi)] \tag{23}$$

In which

$$\varepsilon = \frac{1}{2}(1 + \frac{\delta_a \tan\alpha}{\delta_r}) \tag{24}$$

It should also be apparent that

$$Q_\psi = Q_{a\max}[1 - \frac{1}{2\varepsilon}(1-\cos\psi)]^{1.5} \tag{25}$$

For static equilibrium to exist, the summation of rolling element forces in each direction must equal the applied load in that direction.

$$\begin{cases} F_r = \sum_{\psi=-\psi_1}^{\psi=\psi} Q_\psi \cos\alpha\cos\psi \\ F_r = \sum_{\psi=-\psi_1}^{\psi=\psi} Q_\psi \sin\alpha \end{cases} \tag{26}$$

In which Ψ_1 is the limiting angle defined as follow,

$$\psi_1 = \cos^{-1}(-\frac{\delta_a \tan\alpha}{\delta_r}) \tag{27}$$

Using the integral form of J_r (ε) and J_a (ε) introduced by Sjoväll, Equation (26) may be written in equations system form.

$$\begin{aligned} &\begin{Bmatrix} Fr \\ Fa \end{Bmatrix} = ZQ\max \begin{Bmatrix} J_r(\varepsilon)\cos\alpha \\ J_a(\varepsilon)\sin\alpha \end{Bmatrix} \\ &= ZK(\delta_a \sin\alpha + \delta_r \cos\alpha)^{1.5} \begin{Bmatrix} J_r(\varepsilon)\cos\alpha \\ J_a(\varepsilon)\sin\alpha \end{Bmatrix} \end{aligned} \tag{28}$$

where J_r (ε) and J_a (ε) are defined as follows,

$$\begin{cases} J_r(\varepsilon) = \frac{1}{2\pi}\int_{-\psi_1}^{\psi_1}[1-\frac{1}{2\varepsilon}(1-\cos\psi)]^{1.5}\cos\psi d\psi \\ J_a(\varepsilon) = \frac{1}{2\pi}\int_{-\psi_1}^{\psi_1}[1-\frac{1}{2\varepsilon}(1-\cos\psi)]^{1.5} d\psi \end{cases} \tag{29}$$

The values of the integrals of Equation (28) can be get using Simpson Integral Method, Fig.3 gives the values of J_r (ε) and J_a (ε).

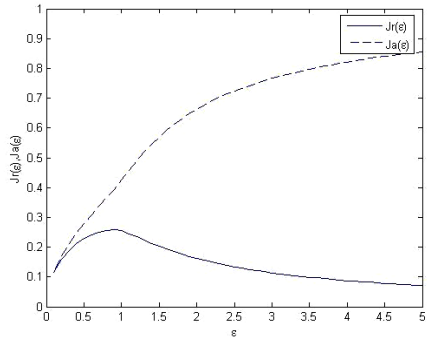


Figure 3: J_r (ε) and J_a (ε) vs. ε for angular contact ball bearing.

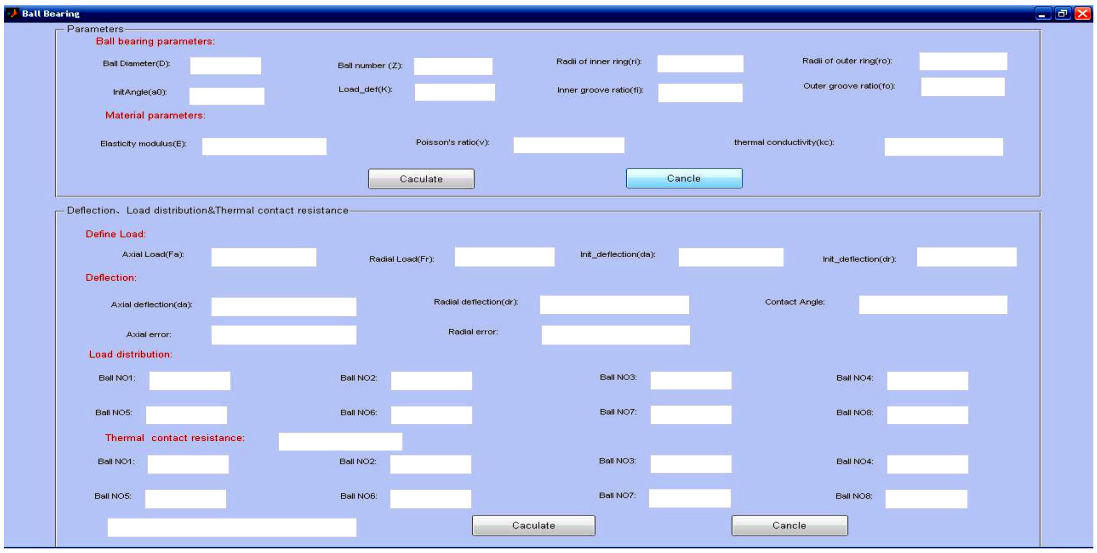


Figure 5: interface of calculation software for thermal contact resistance.

The nonlinear equations system has to be solved by iteration, so the Newton-Raphson method can be applied. When the axial deflection δ_a and the thrust deflection δ_r is determined, the contact force on each ball can be calculated by

$$Q_\psi = K\delta_\psi^{1.5} = K(\delta_a \sin a + \delta_r \cos a \cos\psi)^{1.5} \qquad (30)$$

3.3 Contact Force Under Radial Load

Considering the structure of angular contact balling bearing, when subjected to purely radial load F_r, the normal force Q_i of the rolling element can be decomposed into radial load component Q_{ir} and axial load component Q_{ia} (as shown in Figure 4). The sum of every axial load component was called derivative axial force S, which can be calculated as follows:

$$S = 1.25F_r \tan\alpha \qquad (31)$$

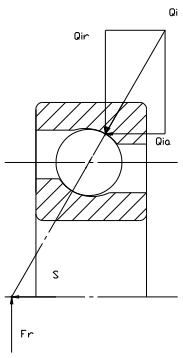


Figure 4: Derivative axial force.

To summarize, when rolling bearing is subjected to purely radial load, an additional derivative axial force is brought out. In this situation, the bearing can be treated as being subjected to simultaneously to a radial load and a centric thrust load.

4 CALCULATION SOFTWARE AND AN EXAMPLE

4.1 Calculation Software

A calculation software has been made using the MATLAB/GUI, whose interface is shown in Fig. 5 below.

The calculation procedure of thermal contact resistance of ball bearing is as follows:

1) Input following parameters of the ball bearing: ball diameter, ball number, radii of inner and outer ring, ratio of inner and outer groove, and the material properties, such as the modulus of elasticity, Poisson's ratio and the thermal conductivity.
2) Calculate the initial contact angle and the load-deflection coefficient of the bearing, which are useful in the calculation.
3) Define the loads of the bearing, and then the load form is analysed.
   a) If the radial load F_r=0, a supposed axial deflection value is needed.
   b) If both the axial and radial loads are positive, supposed axial and radial deflection values must be input for the

calculation.

4) The axial/radial deflection value and the final contact angle are calculated.
5) Finally, the normal load and thermal contact resistance of each ball are obtained.
6) The overall thermal contact resistance of the bearing can be get by connecting the thermal resistance of each ball in parallel.

Take the following bearing as an example: the bearing has 7 spherical balls, and all elements are made from steel 440C; the diameter of the balls, $2r_b$, is 9.525 mm ; the groove radii r_i' and r_o' are $1.03937r_b$, and the race radii r_i and r_o are $3.06037r_b$ and $5.06562r_b$, respectively; the inner and outer bearing diameters are 22mm and 56mm.

4.2 Contact Force Under Centric Thrust Load

In this case, the bearing is subjected to varied thrust load (F_a) ranging from 20 to 200 N with a span of 10 N. The calculated thermal contact resistances are shown in Figure 6.

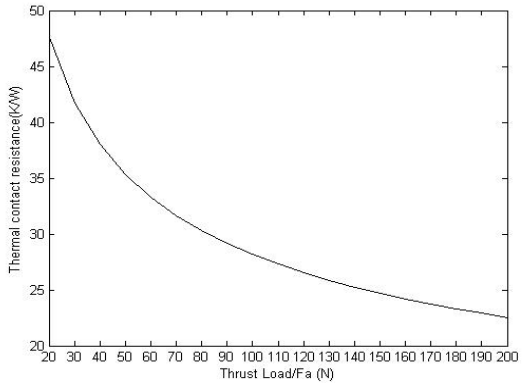


Figure 6: Thermal contact resistance under thrust load.

From Figure 6, we can see with the thrust load increasing, the thermal contact resistance decreases. That is because when the thrust load increases, the normal load of each ball increases, then the contact area extends.

4.3 Contact Force Under Combined Radial and Thrust Load

In this case, the bearing is subjected simultaneously to radial load of 20, 50, 80, 120 N, and varied thrust load ranging from 20 to 200 N with a span of 5 N. The calculated thermal contact resistances are shown in Figure 7.

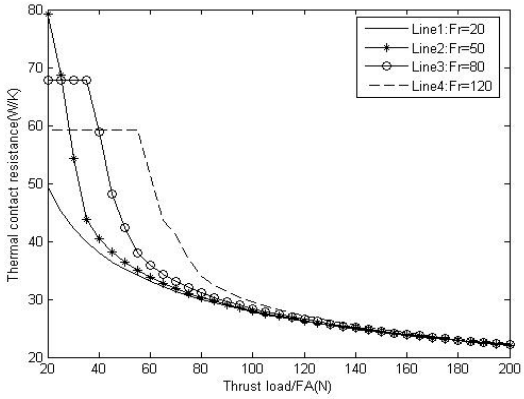


Figure 7: Thermal contact resistance under combined thrust and radial load.

From Figure 7, we can see Line1 represents much the same way as the line in Fig.4; While for Line2, when the thrust load FA increases from 20 to 40 N, the thermal contact resistance changes rapidly. That's because the number of balls subjected to normal load Q_i changes from 1 to 4; For Line3, when the thrust load FA changes from 20 to S (derivative axial force, for F_r=80N, S=38.5N), because FA is smaller than S, the thrust load of the bearing F_a keeps F_a=S, and the thermal contact resistance stays at 67.76 W/K, with only 1 ball subjected to normal load Q_i. The following part of Line3 represents the same way as Line2; Line4 is like Line3.

In another way, for the thrust load FA=100N, the contact resistance of each line is 27.80, 28.00, 28.41, and 29.54, respectively. For combined thrust load FA=100N and F_r=20N, the contact resistance of each ball (as shown in Fig.6) is 185.19, 188.53, 196.74 and 204.17, respectively; for FA=100N and F_r=50N, that is 173.85, 181.04, 201.08 and 223.07; for FA=100N and F_r=80N, that becomes 164.55, 174.68, 206.62 and 250.58; and for FA=100N and F_r=120N, that is 173.85, 181.04, 201.08 and 223.07, respectively.

As mentioned above, the total contact resistance is calculated by connecting that of every ball in parallel. Since the load distribution of Line1 is much more uniform than the others, the total contact resistance of Line1 is smaller.

5 EXPERIMENTAL RESULTS AND COMPARISON

A test program was conducted in order to verify the prediction method of the thermal contact resistance.

Table 1: Experimental results and predictions.

Test Case no.	Load, N		Temperature, K		Q, W	R_{io} K/W	R_s K/W	Measured, R K/W	Predicted, R K/W
	Thrust	Radial	inner	outer					
1	39.2	0	298.2	334.8	0.88	41.59	2.2	39.39	38.24
2	0	98.1	300.3	336.8	0.54	67.3	2.2	65.1	63.30
3	39.2	39.2	300.3	329.1	0.68	42.60	2.2	40.40	39.44

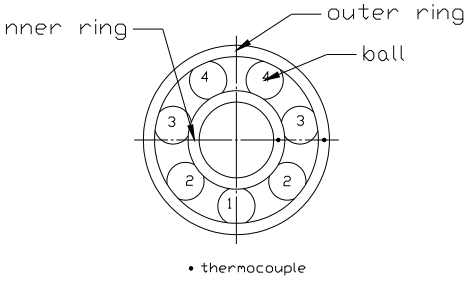


Figure 8: Ball positions and temperature .measurement points (Supposing the balls locate symmetrically).

The experimental apparatus is shown in Figure 9, using thermocouples as sensors. The dimensions and material properties of the bearing matched those described previously. The thrust and radial load was imposed by adjusting the pressure of the hydraulic devices. Figure 8 and 10 show the temperature measurement points on the bearing and the shaft. The test parameters were shown in Table. The temperatures shown in Table were obtained under steady-state conditions when temperature changed less than ±0.2℃.

In the experiment, the bearing was considered as the heat source. The conductive heat flow through the shaft Q was calculated by,

$$Q = Sk_s\left(\frac{T_1 - T_2}{l_{12}} + \frac{T_3 - T_4}{l_{34}}\right) \tag{32}$$

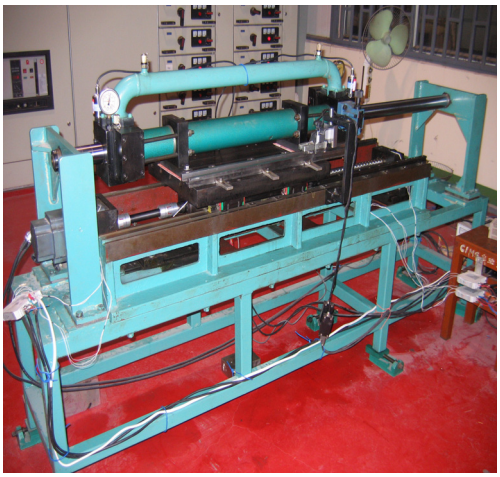


Figure 9: High-speed experimental bench for thermal contact resistance measurement.

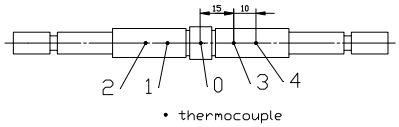


Figure 10: Temperature measurement points on shaft.

Where T_i, T_o and R_s denoted the measured temperatures of the inner and outer rings, and the conductive resistance for the solid existing between the measurement points.

A comparison of the test result and the prediction values is given in Table 1, and the agreement between both results is excellent. Therefore, we can say that the calculation method is applicable to the prediction of contact resistance between the elements of a angular contact ball bearing sustaining thrust, radial and combined loads.

6 CONCLUSIONS

A calculation method based on precisely determined contact forces has been presented to predict the thermal contact resistance between the balls and the inner and outer rings of a space-use dry bearing. The study assumed that a stationary ball bearing sustained axial, radial, or combined loads under a steady-state temperature condition. While the thermal analysis method is the same as that employed to determine constriction resistance, the assumptions commonly utilized in the constriction problem have been numerically confirmed to be applicable to the prediction of the contact resistance between the bearing elements. Also, the calculation of the contact resistance has indicated that the careful consideration of changes in the contact angle is important to determine the contact force and area due to the axial loads.

For the load types dealt with, limited test data were used to verify the proposed method because it was not easy to get the same temperature distribution across the bearing when the magnitude of load was changed, and the total number of operations had to be restricted to avoid changing the surface condition. However, it can be said that the

excellent agreement between the test results and the predictions has confirmed the applicability of the proposed calculation method.

ACKNOWLEDGEMENTS

The work is supported by the National Natural Science Foundation of China (No. 50675076), the National Key Basic Research Special Found of China (No.2005CB724100) and National Natural Science Foundation of China (No. 50575087).

REFERENCES

Yovanovich, M. M., 1975. Thermal Constriction Resistance of Contacts on a Half-Space: Integral Formulation, *AIAA Paper 75-708.*

Clausing, A. M., Chao, B. T., 1965. Thermal Contact Resistance in Vacuum Environment, *Transactions of the American Society of Mechanical Engineers, Ser. C., Vol. 87: 243-251.*

Madhusudana, C. V., Fletcher, L. S., 1986. Contact Heat Transfer—the Last Decade, *AIAA Journal, Vol. 24, No. 3:510-523*.

Yovanovich, M. M., 1986. Recent Development in Thermal Contact, Gap and Joint Conductance Theories and Experiment, *Proceedings of the 8th International Heat Transfer Conference (San Francisco, CA): 35-45*.

Yovanovich, M. M., 1986. Thermal Contact Resistance Across Elastically Deformed Spheres, *Journal of Spacecraft and Rockets, Vol.4, No. 1: 119-12*.

Yovanovich, M. M., 1986. Thermal Constriction Resistance Between Contacting Metallic Paraboloids: Application to Instrument Bearings, *AIAA Paper 70-857.*

Love, A. E. H., 1944. *A Treatise on the Mathematical Theory of Elasticity*, Dover, New York, 4th edition.

Cooper, D. H., 1969. Hertzian Contact-Stress Deformation Coefficients, *Journal of Applied Mechanics*, Vol. 36: 296-303.

Harris, T.A., 2001. *Rolling Bearing Analysis*, John Wiley & Sons, Inc., New York, 5th edition

A NEW APPROACH OF THE NEURAL PREDICTIVE CONTROL APPLIED IN A THERMOELECTRIC MODULE

Adhemar de Barros Fontes, Pablo Amorim and Márcio Ribeiro da Silva Garcia
Departamento de Engenharia Elétrica, Universidade Federal da Bahia, Rua Aristides Novis 2 Federacão, Salvador, Brazil
adhemar@ufba.br, pablo_ssa@hotmail.com, lzeppelin@uol.com.br

Keywords: Neural Control, Neural Networks, Nonlinear systems, Predictive Control.

Abstract: This article presents an efficient solution for predictive control based on neural networks with feedfoward multilayer, as a model for a thermoelectric module. It is shown the capability of a neural network to learn the entire nonlinear dynamics and the advantage of using these nonlinear models for the calculation of the predicted variables. It is also suggested a new control law capable of minimize the cost function using the Newton-Raphson and the descendent gradient optimization rules. For this application it is shown that a significant reduction in the number of iterations and application in real-time systems when compared to other optimization techniques.

1 INTRODUCTION

The recursive neural networks have shown to be a very important tool in several control applications in nonlinear dynamic systems. This article has the objective os presenting a development and its results obtained from a prediction-based algorithm through a recursive neural network applied in a thermoelectric chamber with nonlinear dynamics.

The Generalized Preditive Controller (GPC) was introduced by (Clarke, 1994) and is being used for control of industrial process with profitasble performance in the control of non-minimum phase plants, unstable plants or plants with unknown dead time (Clarke, 1994). The GPC uses initially a linear prediction model. If a nonlinear model is used then it is also necessary to make use of a nonlinear algorithm. Expressive results are obtained regarding efficiency and computational performance. The prediction feature of the GPC when using a neural model, with the capability of learning the entire dynamic of the plant, it is more efficient than the standard nonlinear modelling techniques. It is well known that the plant model is directly related to the accuracy of the prediction (Fontes et al., 2008). The most used techniques to modelling nonlinear plants, as the linearization around the operating points (Lee and Ricker, 1994; Li and Biegler, 1988) or approximated models, do not guarantee the required accuracy when compared to the neural models.

The control signal update rule in the present work is based on the first and second derivatives of the cost function. A new hybrid control law is proposed based on the Newton-Raphson rules and the decrecent gradient. The computational cost related to the computation of the control signal is associated to the Hessian. However, the reduced number of iterations guarantee an excelent performance of the algorithm when it is applied in real-time control systems.

For simulation and real trials of the proposed control technique a Peltier cell was mounted in a set named the thermoeletric chamber. The results obtained caracterize a suitable solution for the proposed system, as well as the potential of this tool in the modelling and control of nonlinear systems, whenever it is real-time or not. In the section 2, the structure of a recursive neural network is presented as well as the equations that describe it. In the section 3, the necessary equations for the determination of the proposed control law. The section 4 presents the description and modelling of the system as well as the results obtained in the trials.

2 NEURAL NETWORK ARCHITETURE

The model of a plant used in a neural generalized predictive controller (NGPC) is a neural network and it is important to evaluate the architeture of the network to be used. It is known that a recursive neural net-

work with three layers, one of them hidden, is capable of representing any linear or nonlinear function. The Figure 1 describes a multilayer feedforward neural network with all its structure of delayed inputs and outputs signals. In (Hao et al., 1993), it was presented in details the mathematical representation of nonlinear plants and the respective considerations in order to determine the order of the neural model in terms of the regressors of the inputs, disturbances and outputs, namely, $u(n)$, $t(n)$ and $y(n)$. For each perceptron of the hidden layer there is an activation function. The neural network is described by the following equation:

$$y(n+1) = f(y(n), y(n-1), \ldots, y(n-n_y), t(n), \ldots, t(n-n_t), u(n), \ldots, u(n-n_u)), \quad (1)$$

or, in a detalied view,

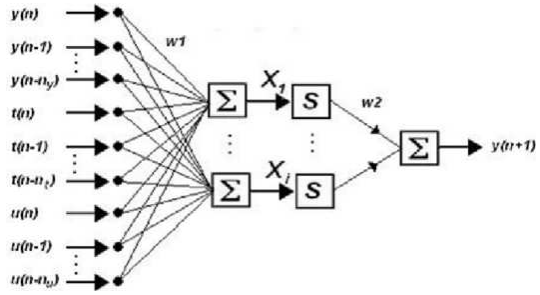


Figure 1: The recursive neural model structure.

$$y(n+1) = \sum_{i=1}^{N} w_2(1,i) S(X_i), \quad (2)$$

with,

$$X_i = \sum_{j=1}^{n_y} w_1(i,j) y(n-j) + \sum_{j=0}^{n_t} w_1(i, n_y+1+j) t(n-j) + \sum_{j=0}^{n_u} w_1(i, n_y+n_t+2+j) u(n-j), \quad (3)$$

where:

- $y(n+1)$ is the output of the neural network;
- $S(.)$ is the output function of the $i-th$ nodes of the hidden layer;
- N is the number of nodes in the hidden layer;
- n_y is the number of inputs nodes associated to $y(.)$;
- n_t is the number of inputs nodes associated to $t(.)$;
- n_u is the number of inputs nodes associated to $u(.)$;
- $w_1(i,j)$ represent the weights associated of the $j-th$ input to the node j;
- $w_2(1,i)$ represent the weights associated of the $i-th$ hidden node to the output node;
- $y(.)$ represents the output past values;
- $t(.)$ represents the ambient temperature past values;
- $u(.)$ represents the input past values.

3 OPTIMIZATION

For the presented application it is necessary the use of a cost function with finite prediction horizon. The NGPC algorithm must satisfy the following optimization problem:

$$\Delta u(k) = \min_{\Delta u} \left\{ J = \sum_{i=N_1}^{N_2} ||ref(n+i) - \hat{y}(n+i)||_2 + \lambda \sum_{i=1}^{N_u} ||\Delta u(n+i)||_2 \right\}, \quad (4)$$

where:

- N_1 is the minimum prediction horizon;
- N_2 is the maximum prediction horizon;
- N_u is the control horizon;
- $ref(n+i)$ is the reference signal;
- $\hat{y}$ is the output signal predicted by the neural model;
- λ is the weight at the control signal;
- $\Delta u(n+i)$ is the variation of the control signal defined as $u(n+i) - u(n+i-1)$.

For the cost defined in (4) there are four tuning parameters: N_1, N_2, N_u and λ. As the dead time is much smaller than the sampling time, it is fair to admit $N_1 = 1$ and, as a consequence, $N_2 = N_u$. It may be verified also that, while the cost is minimized, the resulting control signal allows the plant to track the reference signal.

As mentioned before, the proposed control law presents an update algorithm based on the rules of Newton-Raphson and the decrecent gradient. The use of two methods allows the adding of the weight α, according to (5). This way, the control law contemplates one more degree of freedom and as a consequence a gain in the dynamics of the controller. The control law proposed for the update of the control signal $U(k+1)$ is given by,

$$U(k+1) = U(k) - \alpha\frac{\partial J}{\partial U} - (\frac{\partial^2 J}{\partial U^2})^{-1}\frac{\partial J}{\partial U} \quad (5)$$

It is easily observed that there's an addition of a new degree of freedom α, besides the one defined in (4). The optimal value of J in terms of U is obtained after each iteration of the optimization algorithm. In the iterative process, for each value of J the future inputs vector $U(k)$ is also calculated, with the Jacobian and Hessian matrices defined as:

$$\frac{\partial J}{\partial U} = \begin{bmatrix} \frac{\partial J}{\partial u(n+1)} \\ \frac{\partial J}{\partial u(n+2)} \\ \cdots \\ \frac{\partial J}{\partial u(n+N_u)} \end{bmatrix}, n = 1, \ldots, N_u \quad (6)$$

$$\frac{\partial^2 J}{\partial U^2} = \begin{bmatrix} \frac{\partial^2 J}{\partial u(n+1)^2} & \cdots & \frac{\partial^2 J}{\partial u(n+1)\partial u(n+N_u)} \\ \cdots & \ddots & \cdots \\ \frac{\partial^2 J}{\partial u(n+N_u)\partial u(n+1)} & \cdots & \frac{\partial^2 J}{\partial u(n+N_u)^2} \end{bmatrix}, \quad (7)$$

and the predicted output is given by:

$$\hat{y}(n+k) = \sum_{i=1}^{N} w_2(1,i)S(X_i). \quad (8)$$

The iterations are interrupted when the percentage of the variation $U(k)$ is smaller than a given ε. For each iteration k the elements of the Jacobian and the Hessian are calculated for the given control law.

It is necessary that efficient routines are elaborated for the calculation of the gradient, used in the minimization of the cost, when the iterations of the optimization algorithm are used in real-time.

4 TRIALS

4.1 System Description

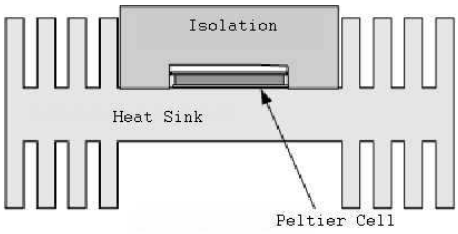


Figure 2: Thermoelectric module.

A Peltier cell, also know as thermoelectric module (TEM),is a component composed of several semiconductors plates placed side-by-side and electrically isolated from the external environment by a ceramic coat. Through two terminals connected to the cell a electric current circles, causing the heat to pump between its faces. This phenomenon is know as the *Peltier efect* and it is the opposite of the *Seebeck efect*, which caracterizes the thermocouples. The heat pumping direction depends on the direction of the current flow, which allows the Peltier cells to be used as actuators in cooling systems as much as in heating systems.

The TEM is widely used in temperature control in specific applications (Fontes et al., 2008). In some of them a physical model is used, (Fontana, 2001; Almeida, 2003). These applications vary from cooling modules, medical instruments composition and small refrigerators, etc (Mel, 1999).

A thermoelectric module is composed of a thermic chamber, where the Peltier cell is associated to a heat sink. The Figure 2 presents an illustration of the set used in the experiments of heat pumping. The process variable is the upper face temperature, controlled by the manipulation of the applied current. The information of the ambient temperature is also informed to the neural net as a load disturbance. Other works involving the thermic chamber present a caracterization of the module by nonlinear parametric models (Sobrinho et al., 2006; Lima, 2007; Almeida, 2004) and bilinear state-space models (Garcia, 2008).

4.2 Neural Network Training

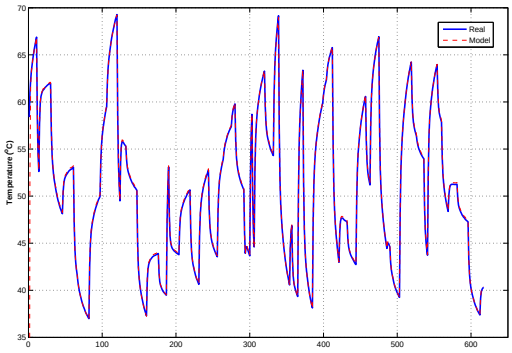


Figure 3: Neural model plot versus system's real data.

For the network trainning, the system response for a Pseudo-random signal input (PRS) was obtained along the entire operation range $(30^oC$ a $80^oC)$. The proposed architeture for the neural model contemplates 3 layers, with 3 regressors in the hidden layer and one in the output layer, 3 exogen regressors and the Hyperbolic Tangent as the activation function;

During the training phase the weights were adjusted iteratively, as proposed in (Hagan and Men-

haj, 1994). For the feedforward architeture proposed, the performance was measured in acordance with the minimization of the squared error criterion between the network and the system's real outputs for the same input signal. Several training algorithms were used, as the resilient backpropagation, the Fletcher-Reeves rule with backpropagation of the conjugated gradient, the Polak-Ribiere rules, the decrescent gradient, among others. The algorithm of Levenberg-Marquardt presented the best overall results and it was used for the system modeling. The Figure 3 presents the output of the neural model compared to the real system's output.

For validation, another PRS was generated and applied to the system and its response was compared to the model response. *The mean squared error* between the system (y_r) and the model $(\hat{y}_m)$ outputs for the N points, defined as

$$E_{mq}(\%) = \frac{\sqrt{\sum_{i=1}^{N}(y_r - \hat{y}_m)^2}}{N} * 100, \qquad (9)$$

was $1,148\%$

The closed-loop system was implemented according to the well-known *receiding horizon principle* ((Propoi, 1963; Camacho and Bordons, 2004; Zamarreo and Vega, 1999)). It is fair to say that the good computational performance of the proposed controller is based on the choice of the algorithm defined in the optimization block. The choice of the controller parameters can be done by several criteria: Number of iterations for the resolution of the control signal, computational cost and accuracy of the solution. It is necessary though, to develop fast optimization algorithms. In this works a solution is presented, which applies the Newton-Raphson method added by the decrecent gradient which can be implemented as optimization technique in highly nonlinear real-time applications. A computational analysis showed that there was no significant increase in the computational cost associated with the optimization of the control signal with the increment of the decendent gradient method. The terms of the decrecent gradient to be calculated are the same as the onse of the Newton-Raphson's method.

4.3 Results

The trials were performed for three distinct cases: The algorithm 01 represents the system proposed in this article with parameters $\lambda = 2000$ and $\alpha = 0.00008$. The algorithm 02 refers to the same algorithm with a more agressive tuning ($\alpha = 0.0000955$). The algorithm 03 refers to an existing algorithm, without the addition of the term related to the decrecent gradient, with $\lambda = 2000$. The prediction horizon is $Ny = 10$. The small values of α are due to the system's large static gain, but the variation of these parameters between the algorithms 1 and 2 is 20%.

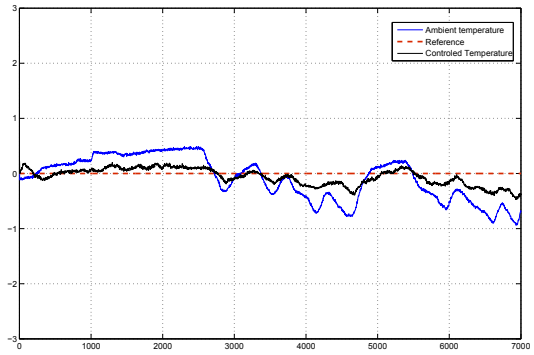


Figure 4: System response to load disturbances.

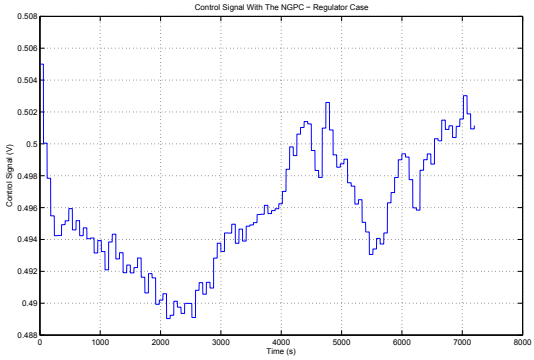


Figure 5: Control signal for the regulator case.

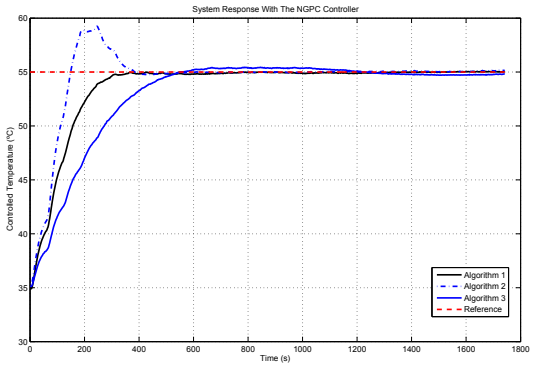


Figure 6: System response for a step in the reference.

The Figure 4 shows the system response to a disturbance caused by the variation of the ambient temperature. It is easy to observe that the controller is capable of rejecting the distubance, keeping the system

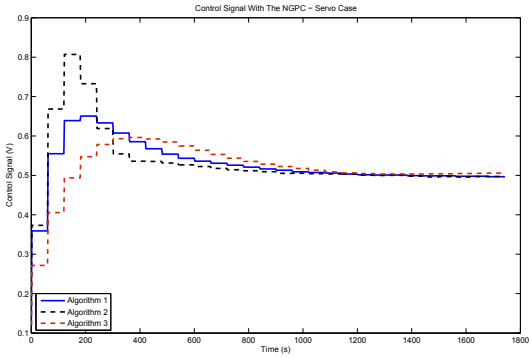


Figure 7: Control signals for the servo case.

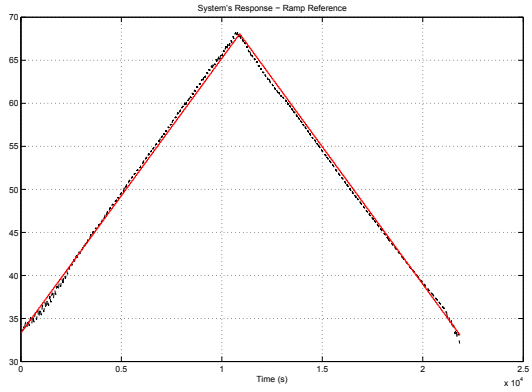


Figure 8: System response for a ramp reference.

output inside the desireble range, presenting a very small offset of 0.3^oC, inside the accuracy range of the sensors, for an ambient temperature variation of 1.5^oC. The output signal of the controller for this case is shown in the Figure 5.

The system response for a deviation of 20^oC in the setpoint for the three controllers implemented is shown in the Figure 6, in which the dashed line represents the system output for the algorithm 02, while the black and blue lines represent the algorithms 01 and 03 respectivaly. The control signal for the three algorithms are expressed in the Figure 7.

For better evaluation of the performance of the controllers, the indices presented in (Goodhart et al., 1994) were used, which are the mean and variance of the controller outputs ($\bar{u}$ and σ_u, respectivaly), the *Integral of the Absolute Error* (IAE), which penalizes the error between the reference and the system variable and the *Integral with Time of the Absolute Error* (ITAE), which penalizes the absolute error throughout the time line. The calculated indices are shown in the Table 1 and the time parameters of the closed-loop system is presented in the Table 2. It is easy to see that the algorithm proposed in the present work presents a better performance in relation to the tracking error and a smaller mean control effort($\bar{u}$). The settling time is also smaller. The rise time for the second algorithm is smaller, but it also presents a large overshoot which may not be viable in real applications. For the regulatory trial, the results obtained were $E_{qm} = 0.0755\%$, $IAE = 0.104$, $ITAE = 145.108$.

The third and last trial was the analyzis of the system response for a ramp reference. The control law does not contemplate a second integral action, necessary for the ramp tracking. Still, the algorithm provided good results when compared to other techniques. In this work, the proposed methodology was compared to three other results obtained for different controllers for the same case: The algorithm B is a single model based GPC presented in (Lima, 2007). The algorithms C and D are described in (Santana, 2008) in a multi-model environment, with the controllers based on gain margin and phase margin metrics, respectively. The algorithm A is the one proposed in this work, with $Ny = 2$, $\lambda = 3000$ and $\alpha = 0.000095$. The ramp started in the temperature of 33.5^oC covering all modeled range with variation of 0.1915^oC at each 60 seconds, which corresponds to one sampling time. The overall results are shown in Table 3

Table 1: Performance indices for the controllers.

	Alg. 01	Alg. 02	Alg. 03
$\bar{u}$	0.5295	0.5332	0.5196
σ_u	0.0031	0.0063	000038
IAE	0.6728	0.6226	1.1550
ITAE	77,1668	81,7647	244,9661

Table 2: Time indices for the controllers.

	Alg. 01	Alg. 02	Alg. 03
Rise Time	200s	118s	358s
Settling Time	296s	354s	972s
Max. Overshoot	0%	21.66%	2.21%

Table 3: Mean squared error for a ramp reference.

	Alg. A	Alg. B	Alg. C	Alg. D
Eqm	0.3191%	2.214%	1.482%	1.418%

The neural networks is capable of capturing the entire dynamic of the system. Performance indices based on the number of iterations and the computational effort when only the jacobian is used was presented in (Soloway and Halcy, 1996). It is fair to affirm that the reduced number of iterations performed by the NGPC algorithm ranges from 6 to 12 times

faster, even with the calculation of the Hessian requiring a high computational effort.

5 CONCLUSIONS

In this work a new approach of the Neural GPC was presented which complies a little modification in the control law, given by the adding of one more degree of freedom associated to the decrecent gradient. This modification caused a significant improvement in the control effort and in the general system's closed-loop response without significant increase of the computational effort. Furthermore, the algorithm becomes much more flexible when compared to other one-degree of freedom based strategies.

The trials were executed in a real-time nonlinear physical system with complex dynamics, with non-minimum phase states and highly nonlinear static gains along the diferente operation ranges. The neural model was able to represent with very good accuracy the system dynamics, which shows the efficiency of the neural networks when applied in the nonlinear system identification. The system was implemented to prove in practice the superior performance of the proposed technique. The same algorithm may be applied in the control of important industrial process, as in multivariable control, level, concentration and temperature. The developed algorithm may estimulate new applications involving the ideas presented in this work. The low computational cost allows the practical implementation of the proposed algorithm in real-time existing embedded systems.

REFERENCES

(1999). *Theromeletric coolers and accessories*. Melcor Corporation, first edition.

Almeida, L. A. L. (2003). *Modelo de Histerese para Transicão Semicondutor-Metal em Filmes Finos de VO2*. PhD thesis, Universidade Federal de Campina Grande.

Almeida, L. A. L. (2004). Modelo dinmico não-linear para módulo termoelétrico. In *XV Congresso Brasileiro de Automtica*.

Camacho, E. F. and Bordons, C. (2004). *Model predictive control*. Springer-Verlag Limited.

Clarke, D. W. (1994). *Advanced in model-based predictive control*. Oxford Univesity Press.

Fontana, M. (2001). Caracterizacão e modelagem das propriedades ópticas de sensores de dióxido de vanádio. Master's thesis, Universidade Federal de Campina Grande.

Fontes, A. B., Dorea, C. E. T., and Garcia, M. R. S. (2008). An iterative algorithm for constrained mpc with stability of bilinear systems. In *16th Mediterranean Conference on Control and Automation*, pages 1526–1531.

Garcia, M. R. S. (2008). Controle preditivo por realimentacão de estado de sistemas bilineares sob restricões aplicado a um módulo termoelétrico. Master's thesis, Universidade Federal da Bahia.

Goodhart, S. G., Burnham, K. J., and James, D. J. G. (1994). Bilinear self-tuning control of a high temperature heat treatment plant. *IEE Proc.-Control Theory Appl.*, 141(1):12–18.

Hagan, M. T. and Menhaj, M. (1994). Trainning feedforward networks with the marquadt algorithm. *IEEE Transactions on Neural Networks*, 5(6):989–993.

Hao, J., Tan, S., and Vandewalle, J. (1993). One step ahead predictive control of nonlinear systems by neural networks. In *Proceedings of 1993 International Joint Conference on Neural Networks*.

Lee, J. H. and Ricker, N. L. (1994). Extended kalman filter based nonlinear model predictive control. *Ind. Eng. Chem. Res.*, 33(6):1530–1541.

Li, W. C. and Biegler, L. T. (1988). Process control strategies for constrained nonlinear systems. *Ind. Eng. Chem. Res.*, (27):1421–1433.

Lima, J. S. (2007). Técnicas de controle preditivo bilinear aplicado a um módulo termoelétrico. Master's thesis, Universidade Federal da Bahia.

Propoi, A. I. (1963). Use of lp methods for synthesizing sampled-data automatic systems. *Automn Remote Control*, 24.

Santana, M. J. C. (2008). Controle preditivo multimodelo aplicado a um módulo termoelétrico: Uma nova métrica. Master's thesis, Universidade Federal da Bahia.

Sobrinho, M. O. S. L., Lima, J. S., Souza, V. O. S. T., Fontes, A. B., and Almeida, L. A. L. (2006). Caracterizacão de um módulo termoelétrico por modelo paramétrico bilinear. Congresso Brasileiro de Automtica.

Soloway, D. and Halcy, P. J. (1996). Neural generalized predictive control, a newton-raphason implementation. In *Symposium on Intelligent Control*.

Zamarreo, J. M. and Vega, P. (1999). Neural predictive control. application to a highly non-linear system. *Engineering Applications of Artificial IntelligenceJapan*, (12):149–158.

THE EXTRACTION OF KNOWLEDGE RULES FROM ARTIFICIAL NEURAL NETWORKS APPLIED IN THE ELECTRIC LOAD DEMAND FORECAST PROBLEM

How Artificial Neural Networks Retain Knowledge and Make Reliable Forecasts

Tarcisio R. Steinmetz, Adelmo L. Cechin and Jose V. Canto dos Santos
PIPCA - UNISINOS, Av. Unisinos, Sao Leopoldo, Brazil
trsteinmetz@unisinos.br, acechin@unisinos.br, jvcanto@unisinos.br

Keywords: Rule extraction from Artificial Neural Networks, Fuzzy Set Theory, Principal Components Analysis, Electric Load Demand Forecast.

Abstract: We present a methodology for the extraction of rules from Artificial Neural Networks (ANN) trained to forecast the electric load demand. The rules have the ability to express the knowledge regarding the behavior of load demand acquired by the network during training process. The rules are presented to the user in an easy to read format, such as *IF premise THEN consequence*. Where *premise* relates to the input data submitted to the network (mapped as fuzzy sets), and *consequence* appears as a linear equation describing the output to be presented by the network, should the premise part holds true. Experimentation demonstrates the method's capacity for acquiring and presenting high quality rules from neural networks trained to forecast electric load demand for several amounts of time in the future.

1 INTRODUCTION

One important issue concerning the requirements of proper load demand forecast methods is the ever increasing dependency of electricity supply for today's industrial societies. Hence, the last decades have shown large investments from energy supply companies in order to improve operation security of electric networks and to ensure quality of service of energy supply for the costumers (Ghods and Kalantar, 2008). These objectives could be achieved through the use of a better knowledge of the load demand behavior for the area supplied by energy supply companies. Such knowledge can even be used to guide the company's tactical and strategic decision making within the company's administrative areas.

This work presents a methodology designed for the extraction of rules form Artificial Neural Networks trained to forecast electric load demand for several amounts of time in the future. The rules obtained describe the knowledge acquired by the network during the training phase. The rules provide insight about the load demand behavior for the area where the training data have been gathered (a city, for instance), such as the impact that each of the input variables cause on the load demand, under what circumstances occurs drastic changes in the load demand pattern, among other important information to support tactical and strategic decisions throughout the energy supply company. This paper proceeds as follows: in the next section we discuss some theoretical aspects. Section 3 details FAGNIS, the rule extraction method used in this work. Section 4 demonstrates the methodology proposed for the proper rule extraction from the trained ANNs. Section 5 shows some of the experiments used to validate the method and the results obtained. In Section 6 we finish the document, presenting our conclusions.

2 THEORY

This section deals with the theoretical concepts used in this paper. Fuzzy Set Theory and Principal Components Analysis are used in this work, however, due to space limitations they are not covered here. The reader should refer to (Angelov, 2002) and (Hastie et al., 2009) to read about these topics.

2.1 Electric Load Demand Forecast

The electricity demand or system load encompasses the summation of electric usage at each consumption point (users) supplied by an electric supply facility.

Its behavior is highly dynamic and difficult to comprehend. The amount of variables involved in the characterization of the load demand curve is large indeed, and different effects are perceived by the same variables in different regions of the globe. However, works such as (Gross and Galiana, 1987), (Srinivasan et al., 1995), (Srinivasan et al., 1999) and (Ghods and Kalantar, 2008) show that certain factors are commonly responsible for affecting the load demand.

2.2 Rule Extraction from Trained Neural Networks

The reason for the successful application of AANs in fields as diverse as academia, industry and commerce is its generalization capabilities. However, this high power of generalization comes at a price: it prevents the network from expressing the knowledge acquired during the training phase. Thus the network can be seen as a black box, presenting to the user the predicted output based on the input data, while important knowledge about the problem studied remains encrypted within the network's weight matrix, never to be discovered.

In order to solve this problem, rule extraction techniques can be used to acquire the knowledge embedded in the network's weight matrix, and then to present it to the user in a clear interface. As mentioned before, rules have a IF *premise* THEN *consequence* structure, where *premise* somehow defines the vector of input data presented to the network and *consequence* describes the output to be obtained should the *premise* part holds true. Mostly of the rule extraction methods used today rely on Fuzzy Sets concepts to describe the premise part of the rules (Cechin, 1998), (Benitez et al., 1997).

In an important survey concerning several rule extraction methods, Andrews et al. (Andrews et al., 1995) present several interesting features displayed by transparent neural networks, that is, ANN capable of describing their knowledge to the user. In this work the authors mention that ANNs with explanatory capabilities are capable of (among others): (1) operating in conjunction with symbolic intelligent systems; (2) controlling critical applications such as air traffic control and support scientific theory formulation.

3 FAGNIS - RULE EXTRACTION FROM SIGMOID NETWORKS

This section describes FAGNIS (Cechin, 1998), the rule extraction method selected for this work. FAGNIS has been considered because of its ability to extract rules from standard feedforward neural networks, with or without shortcut connections, and heaving one or more hidden layers. Further, since FAGNIS performs on an already trained network, it has no dependency on its training algorithm. In fact, any training algorithm can be used, from the standard backpropagation to algorithms yet to be created. Requirements such as special ANN architectures and special or adapted learning algorithms are mandatory in the majority of rule extraction methods, namely (Jang et al., 1997) and (Nauck et al., 1994).

FAGNIS begins its extraction procedure by splitting the sigmoid curve within the hidden neurons in three regions. These regions are then transformed in straight lines, which are mapped by very simple equations, as illustrated in Figure 1.

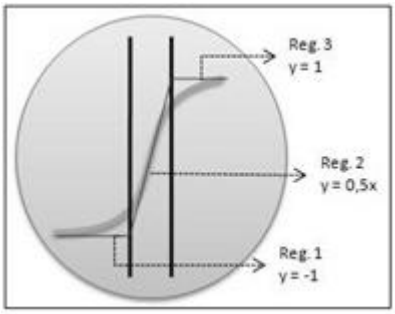


Figure 1: Separation of the sigmoid curve within the hidden neurons performed by FAGNIS.

Next, the training data are once more submitted to the network, where FAGNIS verifies the resulting activation of the hidden neurons for each of the data points. The data points are then grouped according to the activation regions (as shown in Figure 1) generated within the network's hidden neurons.

To assemble the premise part of the rules, FAGNIS transforms each group of data points found in the previous step in fuzzy sets. The fuzzy sets are represented by the midpoint of each group. The consequence part of the rule is defined as a linear equation that represents the output dependence of the network on the input data. The expressions below show two rules acquired from a fictitious neural network

$$\text{IF } (x_1, x_2) \text{ is } G_1 \text{ THEN } y = x_1 w_{ij} + x_2 w_{ij} + k$$

$$\text{IF } (x_1, x_2) \text{ is } G_2 \text{ THEN } y = x_1 w_{ij} + x_2 w_{ij} - k$$

where G_1 and G_2 are fuzzy sets (with membership functions μ_1 and μ_2 respectively), w_{ij} is the weight linking the i-th neuron to the j-th neuron and k is the intercept value for the equation.

4 METHOD

The first step of the process is to prepare the ANN to be used as the forecaster model. Some consolidated techniques, such as variable selection method and cross-validation technique were used to improve the model, as well as to decide on key issues concerning the ANN's architecture. The Mean Absolute Percentage Error (MAPE) metric was selected to measure the ANN's accuracy, as shown in Equation (1):

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{A_i - F_i}{A_i} \right| \quad (1)$$

where n is the number of data points, A_i is the actual value and F_i is the forecast value. Once the dataset is composed, each data point is normalized as shown in Equation (2):

$$nd_i = \frac{d_i - \mu_D}{\sigma} \quad (2)$$

where nd_i is the normalized data, d_i is the actual data (from the dataset), μ_D is the mean of the data column and σ is the standard deviation for the data column.

Principal Components Analysis (PCA) is then executed on the dataset, and the resulting dataset is then used to train the neural network. The principal components are selected based on Jolliffe's criterion (Jolliffe, 2002).

Once the neural network is properly built and trained, FAGNIS can be used to extract knowledge rules from it.

The algorithm is executed as detailed in Section 3. Once the execution is terminated, the rules can be analyzed and interpreted. The equations on the consequence part of the rules explain the load demand behavior for the data assigned to the fuzzy sets described in the premise part. To determine which of the input variables is the most important, that is, the one that has the major influence on the load demand, the user needs simply to identify which of the independent variables of the equation has the highest absolute coefficient value.

5 EXPERIMENTS AND RESULTS

This section describes the results obtained from two experiments used to evaluate the proposed methodology. The first experiment concerns the extraction of rules from ANN trained to predict the average load for the next hour. In the second experiment, the forecast window is expanded to the next month.

5.1 Experiment 1

The load demand forecast for the next hour constitutes a classical problem within this field, and the adoption of neural networks techniques usually leads to excellent results. Regardless of the triviality of this problem, the energy supply companies should not underestimate the value of such information: the load demand for the next hour provides support to several of the company's tactical decisions like the expansion of transmission lines, equipment maintenance schedule and other routine activities.

The dataset used in this experiment corresponds to the hourly load recorded from 2003 to 2007, For a capital city holding approximately 1.4 million of citizens. The data has been arranged in such a way that daily, weekly, monthly and annually load patterns could be learned by the neural network. The list below represents the structure of the dataset prior to PCA application:

- load demand for the last twenty-four hours (24 columns),
- load demand for the forecast hour registered in the last six days (6 columns),
- load demand for the same day and forecast hour registered in the last three weeks (3 columns),
- load demand for the forecast hour and day registered in the last month (1 column),
- dependent variable: average load demand for the next hour.

After data processing, the principal components were extracted. Jolliffe's Criterion informed that the first eight components should be used as the input layer of the neural network. The data belonging to the remaining components were discarded from the experiment.

The neural network selected by tenfold cross-validation method has eight neurons in the input layer, four in the hidden layer and one in the output layer. The network's accuracy, measured by the MAPE metric, is of 0,027%.

Some of the rules extracted by FAGNIS appear in table 1. Not all the thirty-four rules were displayed due to space reasons. The rules show that the three first principal components have an increasing effect on the load (their coefficient values are positive in all the thirty-four rules). On the other hand, the fourth principal component has a decreasing effect on the load demand (it has negative coefficient value in all the rules).

The rules are arranged in the following format:

IF (PC_1 is F_1) AND (PC_2 is F_2) ... AND (PC_8 is F_8)

THEN $y = int + KPC_1 + KPC_2 + ... + KPC_8$

Table 1: Some of the rules found by FAGNIS in experiment 1.

Rule #	Rule description	Data points
1	IF x = (0.319 -1.272 -0.431 -0.091 0.275 -0.349 -0.159 -0.312) THEN y = (0.036 0.189 0.174 0.128 -0.175 0.159 0.047 0.038 -0.093)	8946
2	IF x = (1.913 1.350 0.327 -1.651 -0.609 0.923 0.301 0.317) THEN y = (-0.004 0.208 0.191 0.179 -0.096 0.204 -0.001 0.237 0.131)	3948
3	IF x = (1.179 -0.652 1.299 0.420 0.080 0.006 0.867 0.390) THEN y = (-0.085 0.215 0.207 0.156 -0.183 0.136 0.111 0.192 0.162)	3556

where PC_n is the principal components used as the input layer of the neural network, F_n are the fuzzy sets representing the data being submitted to the ANN's input layer, K are the coefficient values of the linear equation and *int* is the point where the straight line defined by the equation intercepts the Y axis.

Figure 2 depicts the first column of the rotation matrix resulted from PCA application. It says that the first principal component is composed mainly by the load of the forecast hour registered one day ago. Table 2 presents the results of the same analysis for the remaining principal components.

Table 2: Most important variables used for principal components characterization.

PC	Description
1	Load for the forecast hour, 1 day ago
2	Load for the forecast hour, 5 days ago
3	Load registered 24 hours ago
4	Load for the forecast hour, 1 week ago
5	Load registered 14 hours ago
6	Load registered 11 hours ago
7	Load for the forecast hour, 4 days ago
8	Load for the forecast day and hour, 1 month ago

Based on the rules found and the information detailed on table 2, the following assertions can be made:

1. the load demand registered in the last twenty-four hours before the forecast, as well as one day before and five days before have an increasing effect on the load demand
2. the load demand for the same time of the forecast, registered one week ago, decreases the load demand for the next hour

5.2 Experiment 2

The load demand forecast for the next month is a task much more difficult than that of the previous experiment. As the window of forecast expands to such a long time, economic factors begin to play a more important role in shaping the load demand curve (Srinivasan et al., 1999). The load demand for the next month consists in strategic information to energy supply companies. It supports the company to purchase an amount of energy very close to the amount to be used by its costumers, thus increasing the company's profit.

In this experiment, electric demand and climatic data were used to determine the load demand for the next month for a small city with a large number of industries. The data have been stored on a daily basis, for the period of 2005 to 2007. The file structure before PCA is shown below:

- residential load demand registered for 120, 90, 60 and 30 days prior to forecast (4 columns),
- industrial load demand registered for 120, 90, 60 and 30 days prior to forecast (4 columns),
- commercial load demand registered for 120, 90, 60 and 30 days prior to forecast (4 columns)
- average temperature registered for 120, 90, 60 and 30 days prior to forecast (4 columns),
- average relative air humidity registered for 120, 90, 60 and 30 days prior to forecast (4 columns),
- dependent variable: average load demand for the next 30 days.

The first seven principal components were used to build the training dataset to the neural network. The reason for the separation of the load in three different categories relies on the city's economic structure: it is mainly industrial. However, the commerce sector has shown rapid increase in the last decades. Thus, it is expected that the rules found show high dependency of industrial load to forecast the average monthly load for this city.

The neural network selected via tenfold cross-validation has seven neurons in the input layer, thirty-two in the hidden layer and one in the output layer. Shortcut connections were not used. This architecture resulted in a MAPE of 3.26%, however, due to the elevated quantity of hidden neurons, more than 300 rules were extracted. It means that many fuzzy

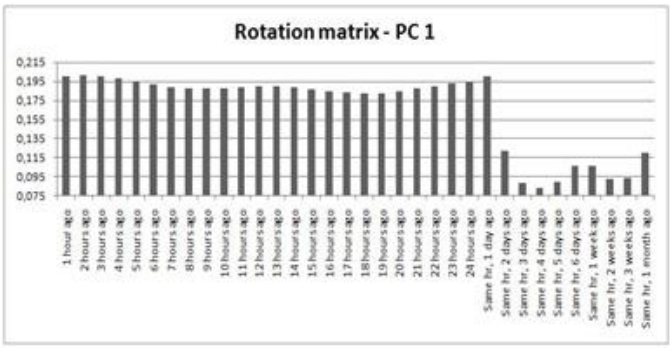


Figure 2: Quantity of information from original dataset used to create the first principal component.

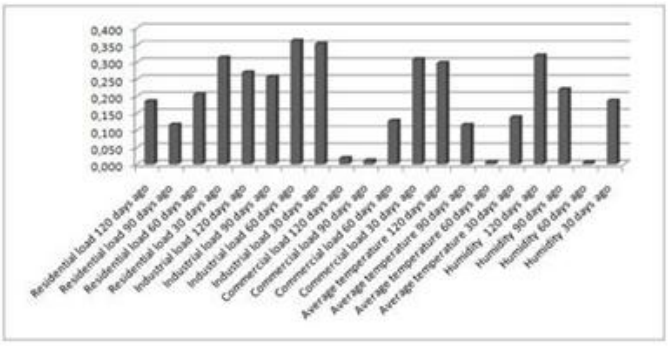


Figure 3: Quantity of information from original dataset used to create the first principal component.

sets were necessary to map the knowledge acquired by the network, and these sets refer to very few data points in the training data. To solve this problem, a new network structure was used in the rule extraction procedure: it has eight hidden neurons, and its MAPE value is of 6.53%. This neural network produced thirty rules.

Table 3 shows the most important rules extracted by FAGNIS. They can be read by the same manner as those shown in the previous experiment. Again, not all the rules could be presented due to space reasons. Figure 3 details the data on the first column of the rotation matrix resulted from PCA application. It shows that the industrial load registered sixty days ago has a strong relation to the load for the next month. However, it is clear that the industrial load registered thirty days ago has also a significant participation in shaping the overall monthly load. Table 4 presents the quantity of information given by the original dataset to build the remaining principal components.

Based on the analysis of the first rule found and the rotation matrix resulted from PCA application (Table 4), it is possible to verify that the industry load demand has high impact on the general monthly load demand of the considered city.

Figure 4 shows the monthly load demand curve for the city. The dots represent the data points associated with rule number 1. This shows that the Fuzzy Sets of the first rule have elevated energy consumption. The analysis described in this experiment can be replicated to the other rules, so that all knowledge learned by the neural network can be acquired.

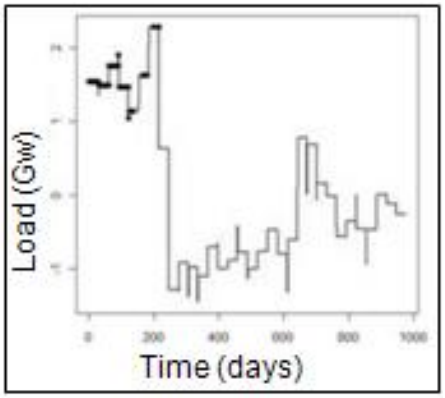


Figure 4: Monthly load demand and data explained by Rule 1.

Table 3: Rules found by FAGNIS in experiment 2.

Rule #	Rule description	Data points
1	IF x = (-2.863 -0.715 1.839 -0.294 0.264 -0.014 0.035) THEN y = (1.442 -0.024 0.001 0.032 -0.033 0.005 -0.004 -0.003)	110
2	IF x = (2.554 -1.616 -1.161 0.165 0.154 -0.095 -0.022) THEN y = (-0.290 -0.002 -0.378 -0.117 -0.392 0.006 -0.005 -0.235)	68
3	IF x = (4.356 0.069 2.064 -0.852 -0.152 -0.221 0.596) THEN y = (-2.804 0.874 0.339 -0.386 0.737 -0.027 0.001 0.127)	42

Table 4: Most important variables used for principal components characterization.

PC	Description
1	Industrial load, 60 days ago and Industrial load, 30 days ago
2	Average temperature, 60 days ago
3	Average temperature, 120 days ago
4	Industrial load, 30 days ago
5	Average humidity, 60 days ago
6	Average humidity, 90 days ago
7	Average humidity, 120 days ago

6 CONCLUSIONS

A methodology for the acquisition of rules from neural networks trained to forecast electric load demand has been presented here. Results found through several experiments (been two of them shown in this paper) attest the methodology's efficiency in extract and present high quality rules for different amounts of time in the future.

Throughout the execution of many experiments, it was made clear that there is a need to differentiate the neural networks of load forecast from those used to rule extraction: the former needs several training cycles in order to obtain a perfect fit to the load demand curve; the latter requires only a few training cycles to obtain the overall knowledge about the load demand, that is, so that a small number of rules can be used to refer to a large quantity of data points.

Both the forecast model and the rules acquired can be used as decision support tools for energy supply companies. For example, several simulations could be used for the executives to better understand load demand behavior in different scenarios, such as future climatic changes.

REFERENCES

Andrews, R., Diederich, J., and Tickle, A. (1995). Survey and critique of techniques for extracting rules from trained neural networks. *Elsevier Knowledge-Based Systems*.

Angelov, P. (2002). *Evolving Rule-based Models: A Tool for Design of Flexible Adaptive Systems (Studies in Fuzziness and Soft Computing)*. Physica-Verlag, Heidelberg, first edition.

Benitez, J., Castro, J., and Requena, I. (1997). Are artificial neural networks black boxes? *Neural Networks, IEEE Transactions on*.

Cechin, A. (1998). *The Extraction of Fuzzy Rules from Neural Networks*. Shaker Verlag, Tubingen.

Ghods, L. and Kalantar, M. (2008). Methods for long-term electric load demand forecasting; a comprehensive investigation. *Industrial Technology, 2008. ICIT 2008. IEEE International Conference on*.

Gross, G. and Galiana, F. (1987). Short-term load forecasting. *Proceedings of the IEEE*.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, New York, second edition.

Jang, J., Sun, T., and Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing. A Computational Approach to Learning and Machine Intelligence*. Prentice-Hall, New Jersey.

Jolliffe, I. (2002). *Principal Component Analysis*. Springer Series in Statistics, New York.

Nauck, D., Klawonn, F., and Kruse, R. (1994). *Neuronale Netze und Fuzzy-Systeme*. Vieweg and Sohn.

Srinivasan, D., Chang, C., and Liew, A. (1995). Demand forecasting using fuzzy neural computation, with special emphasis on weekend and public holiday forecasting. *Power Systems, IEEE Transactions on*.

Srinivasan, D., Tan, S. S., Cheng, C., and Chan, E. K. (1999). Parallel neural network-fuzzy expert system strategy for short-term load forecasting: system implementation and performance evaluation. *Power Systems, IEEE Transactions on*.

NEURO-FUZZY CONTROL OF NONLINEAR SYSTEMS
Application in a Ball and Beam System

Marconi Câmara Rodrigues, Fábio Meneghetti U. Araújo and André Laurindo Maitelli
Departamento de Engenharia de Computação e Automação
Universidade Federal do Rio Grande do Norte, Natal, RN, Brazil
marconi@dca.ufrn.br, meneghet@dca.ufrn.br, maitelli@dca.ufrn.br

Keywords: Control, Hybrid Systems, Fuzzy Systems, Artificial Neural Networks, NEFCON, ANFIS.

Abstract: This study shows both the development and characteristics of some of the main techniques used to control nonlinear systems. Starting from a fuzzy controller, it was possible to apply similar learning techniques to those used in Artificial Neural Networks (ANNs), and evolve to ANFIS and NEFCON neurofuzzy models. These neurofuzzy models were applied to a real ball and beam plant and both their adaptations and their results were discussed. For each controller developed the input variables, the parameters used to adapt the variables and the algorithms applied in each one are specified. The tests were performed in a ball and beam plant and the results are directed toward obtaining a comparison between the initial and final evolution phase of the neuro-fuzzy controllers, as well as the applicability of each one according to their intrinsic characteristics.

1 INTRODUCTION

Controllers such as PIDs and those based on the realignment of states, are simple to implement and inexpensive to operate. However, adjusting their parameters can take considerable time and their performance is usually limited. A number of automatic adjustment techniques for PID controller parameters were developed (Oliveira, 1994; Coelho and Coelho, 1999). These techniques, in addition to raising operational cost, do not enable PID controllers to resolve problems, such as controlling nonlinear systems or maintaining their performance in the presence of uncertainties or parametric variations.

The need for more encompassing techniques in the control area led to the emergence of intelligent techniques such as artificial neural networks, fuzzy systems, genetic algorithms and other techniques based on reinforcement learning.

The fuzzy systems and Artificial Neural Networks (ANNs) are very useful tools for nonlinear systems control, with or without mathematical models. These two techniques will be the main focus of this paper, since it was from their structures that the hybrids presented here were developed.

ANNs attempt to reproduce the capacity of learning and generalizing the knowledge of the cerebral structure of living beings. Based on simple known structures such as artificial neurons, the data propagate from neuron to neuron via synapses, whose weights can be adjusted over the course of the learning process (Haykin, 2001).

Around thirty years after the introduction of the fuzzy set theory by Lotfi Zadeh (Zadeh, 1965), the researcher Jyn-Shing Roger Jang published an article in which fuzzy parameters were calculated using the backpropagation technique, widely used to adjust the synaptic weights of ANNs (Jang and S., 1995). This technique of associating fuzzy with artificial neural networks became known as neuro-fuzzy and the model implemented by Jang was called the Adaptive-Network-Based Fuzzy Inference System, or ANFIS. Other neuro-fuzzy models were also developed and, among these, the NEFCON model (neuro-fuzzy controller) stands out for having easy-to-implement characteristics in real time.

This study was developed because of the potential applicability of ANFIS and NEFCON hybrid intelligent controllers, and showed characteristics of implementation and of use, the intrinsic characteristics of each model, as well as a number of advantages and disadvantages.

This study is divided as follows: section 2, discusses the neuro-fuzzy implementation approaches proposed by the ANFIS (Qiang et al., 2008) and NEFCON (Shujaec et al., 2002) models. Section 3, presents the control structure used to obtain the results of the application, in real time, of the hybrid

techniques in a beam and ball plant. The last section, 4, contains the conclusions about the techniques presented and possible future studies.

2 NEURO-FUZZY CONTROLLERS

Neuro-Fuzzy controllers can aggregate different characteristics of fuzzy controllers and artificial neural networks in a single structure. Thus, the controller based on neuro-fuzzy models will have easy-to-interpret control actions, promoted by the fuzzy controllers, and a learning stage, which is the main characteristic of neural networks. This section presents two neuro-fuzzy controllers, ANFIS and NEFCON.

2.1 ANFIS Model

The ANFIS model uses a fuzzy controller as its basic structure, which can be interpreted as a 6-layer neural network, in which learning techniques such as backpropagation can be applied (Jang et al., 1997).

To simplify, consider a fuzzy system with two inputs (x and y), two membership functions (MFs) for each input variable and one output z. Figure 1 illustrates the structure of the ANFIS model considered.

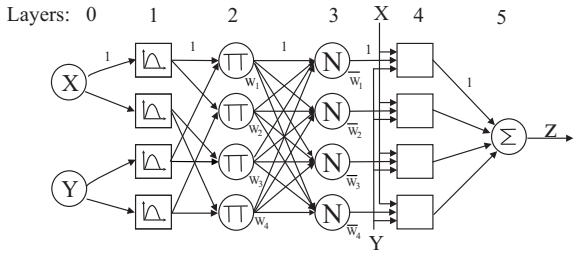


Figure 1: ANFIS with two inputs (x and y) and one output (z).

Data flow can be analyzed layer by layer, as shown below:

- **Layer 0:** Represents the inputs of the model.
- **Layer 1:** The neurons of this layer represent the MFs of the input; that is, the fuzzification phase.
- **Layer 2:** Represents the number of rules. The outputs of the previous layer are operated according to the inference phase of the fuzzy system considered. Multiplication is an interesting option, since it is easy to derive and simple to implement.
- **Layer 3:** The output of this layer will be the output of normalized neurons from the previous layer; that is, the output of each neuron of the previous layer divided by the sum of the output of all the neurons in this same layer.
- **Layer 4:** The function associated to the neurons of this layer will be the function $f(x,y)$, used by the Sugeno model, in which x and y are the system inputs and p, q and r are the adjustable parameters of the Sugeno function.
- **Layer 5:** In this layer the sum of the neuron outputs of the previous layer occurs, obtaining thus the control signal for the system.

From the neural network presented, it can be clearly observed that the neurons that need learning are present in layers 1 and 4, since layer 1 contains the input MFs and layer 4 the Sugeno polynomial, which define the implications of the rules.

The adjustment of controller parameters in the ANFIS model can be obtained using adaptive techniques such as the backpropagation algorithm. The derivatives of equations are found in (Jang et al., 1997). To improve the convergence speed of the algorithm, the η-adaptive technique was used (Rezende and Maitelli, 1999), in addition to backpropagation.

2.2 NEFCON Model

NEFCON is a neuro-fuzzy controller model based on the generic architecture of an ANN, but specifically of a 3-layer perceptron network (figure 2). Making a parallel between the 3-layer perceptron networks and the fuzzy systems, we have (Nrnberger et al., 1999):

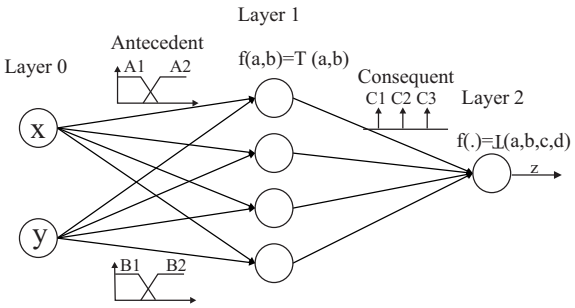


Figure 2: NEFCON with 2 inputs (x and y) and 1 output (z).

- **Layer 0:** This layer represents the inputs. The MFs are found in the synapses that link this layer to the following layer and it is in these synapses that fuzzification occurs.
- **Layer 1:** This layer (intermediary layer) abstracts the fuzzy system rules and it is where the actuation level of each output membership function (MF) is found. In the synapses between this layer

and the following layer, the actuation level found acts on the consequent MFs.

- **Layer 2:** Layer 2 is responsible for defuzzification. The algorithm proposed by (Nrnberger et al., 1999), suggests the use of the mean of the maximums method for the defuzzification stage, which reduces the consequent MFs to simple impulses located in their maximums.

The learning algorithm of the NEFCON model is based on the idea of reinforcement learning (Sutton and Barto, 1998). This algorithm can be divided into two main phases: creating rules and optimizing the MFs (Rodrigues et al., 2004; Rodrigues et al., 2006).

To create rules the algorithm may or not receive a set of initial rules. If a set of initial rules is received, then the rule creation phase will optimize it. It is necessary to specify both the actuation intervals for each input and the actuation interval for the output, since these intervals enable the interval to compare different-sized inputs and create a rule to reduce the input of greatest error.

To illustrate this idea, consider the actuation intervals of two inputs (E1 and E2) are [5, 5] and [-0.5, 0.5], respectively. The input E1 error is 1, and the E2 error is 0.3. To calculate the error of an input in terms of its actuation interval, the value of the input error is divided by the size of the actuation interval. Thus, the error of E1 will be 0.1, whereas the error of E2 will be 0.3. Observe that even though it has a smaller absolute value, the value of input E2 is greater than that of input E1, a situation that leads the algorithm to create a rule with a tendency to minimize the error of input E2. Thus, the algorithm will discover which rule from the set of rules is activated with greatest strength and greatest μ. This rule will activate an output MF that will be obtained by comparing the input in its actuation interval and the output interval divided into output MFs.

Different from that suggested by the algorithm that inspired this work (Nrnberger et al., 1999), a previously created rule can be modified at any moment if a different rule is found for the situation. There is also no need to modify the structure of the MFs, given that they will be treated in a later phase.

To optimize the MFs, the algorithm uses a strategy similar to that used by reinforcement learning. When an input MF is activated, it contributes to reducing or increasing the error. If the action produced provokes an increase in system error, this MF has its actuation field reduced. Otherwise, the MF has its actuation interval increased. A similar situation occurs with the output MFs. However, with these, the gain or loss occurs in their intensity; that is, the MF that collaborates with the increased error will have its intensity reduced; otherwise, the intensity will be increased.

Mathematically, we have that the plant E error is found according to the insertion of inputs into the plant. The contribution, tr, of each rule for the output is estimated and error Er is calculated for each rule unit, according to the following equation,

$$Er = \mu \cdot E \cdot sgn(tr) \tag{1}$$

in which:
$sgn(tr) = tr$ signal
With these data, the consequent modifications can be represented by:

$$\Delta b_i = \eta \cdot \mu \cdot E \tag{2}$$

And the antecedent modifications by:

$$\Delta a_j^{(i)} = -\eta \cdot Er \cdot (b_j^{(i)} - a_j^{(i)}) \tag{3}$$

$$\Delta c_j^{(i)} = \eta \cdot Er \cdot (c_j^{(i)} - b_j^{(i)}) \tag{4}$$

in which:
η = Learning coefficient
a, b, c = Vertices of the membership functions
It can be easily observed that the algorithm does not alter the position of the MF of the antecedents. Its base is only increased or decreased proportionally on both sides; that is, if the MF has a positive contribution, there will be greater likelihood of its occurring again.

A number of restrictions were inserted to avoid the overlapping of more than two MFs and to avoid the emergence of gaps between them. Therefore, an overlap between zero and 50% must be guaranteed; that is, the vertices of a triangle must be contained in the interval corresponding to the middle of the base of neighboring triangles.

3 APPLICATION IN THE BALL AND BEAM SYSTEM

The ball and beam system, in which the controllers were tested, is composed basically of a beam-ball system, a servo motor with a reducer gearbox and a ruler with a ball of reference. There are three sensors, one to measure the position of the reference ball, another for the position of the ball to be controlled and one to measure the angular position of the servo motor (figure 3).

The aim of the controller is to make the ball placed on the beam to follow the pathway specified by the reference ball. Thus, a control system is designed to

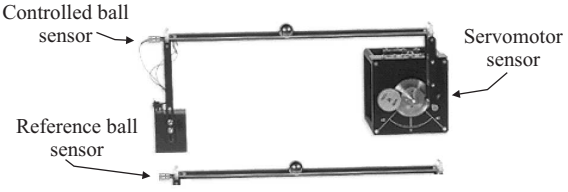


Figure 3: Sensor location in the ball and beam system.

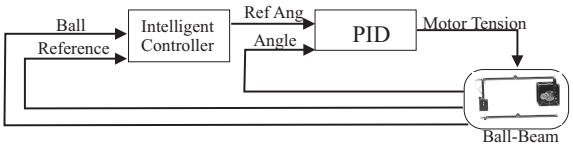


Figure 4: Flowchart illustrating the ball and beam control loops.

send a voltage to the servo motor, which, upon moving, raises or lowers one of the ends of the beam, causing the ball to move.

The control system designed for the ball and beam is divided into two loops (figure 4): one external, where an intelligent controller is responsible for receiving the position of the ball to be controlled and the position of the reference ball and from this, provide the reference angle for the second loop, where a PID controller generates a control signal, which is required for the servo motor to position itself according to the reference angle supplied by the first controller.

The use of the intelligent controller to substitute the two control loops of the plant is a considerable challenge. Thus, it was decided to use hybrid techniques to substitute only the external loop, given that the internal loop functions sufficiently well with the PID designed and the external loop is a sufficient challenge for the proposal.

3.1 Results with ANFIS

The ANFIS model will be optimized by backpropagation. To achieve this, training pairs (points) must be obtained. A PID (Proportional Integrative Derivative) controller was used to obtain the training pairs, substituting the intelligent controller (figure 4). The choice of three inputs for the ANFIS model was based on the characteristics of the controller to be copied, namely the PID. The three inputs considered were the errors: current and previous of the ball to be controlled with respect to the reference ball, and the previous reference angle.

The training point capture stage was followed by the training stage of the ANFIS model. The training algorithm used the input-output pairs obtained with the PID controller to adapt the adjustable parameters of the system.

Seen as a fuzzy system, it has 5 bell-shaped MFs for each input variable. Thus, the intersections between the MFs form 125 possible rules that reference first-order polynomials ($px+qy+rz+s$), as a function of input variables $x =$ current error, $y =$ previous error and $z =$ previous reference angle. The variables p,q,r and s are the adjustable parameters of the polynomials and all were initiated with a value of 0 (zero).

The learning coefficient for this system was initiated with $\eta = 10^{-6}$, the initial values for the MFs were uniformly distributed within the intervals and the sample period considered was 0.05 s.

The initial value for variable η and the intervals in which the MFs were distributed were two of the greatest difficulties in implementing the model. When the η value was elevated (around 10^{-4} for this model) or when the intervals were very different from the reality of the system to be controlled, the program became unstable.

An association with the fuzzy system shows that the equivalent neural network will have the following number of neurons in each layer: 3 - layer 0; 15 - layer 1; 125 -layers 2, 3 and 4; 1 - layer 5.

For training, a total of 800 points were collected from the PID of the external loop of the control system that acts on the ball and beam. After training, the input MFs have optimized shapes, as shown in fig. 5.

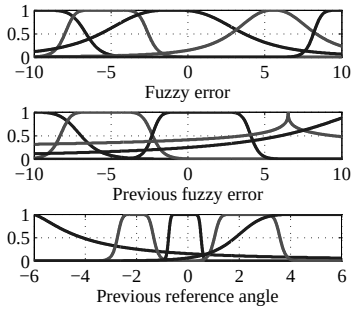


Figure 5: Membership functions after ANFIS adjust.

For an effective learning validation, the controller trained in the ball and beam plant was used for a sequence of preestablished reference points, which dispensed with the use of the reference beam. The effect of the controller on the plant can be observed in figure 6. First, the reference received a pseudo ramdom signal (figure 6). Notice that in the signal changes, the reference is not followed with much perfection by the neuro-fuzzy controller. Analysis of the control signal from ANFIS showed that the beam receives the angle which, in theory, would take the ball to the position of reference. However, the influence of dry attrition causes the ball to be controlled to remain

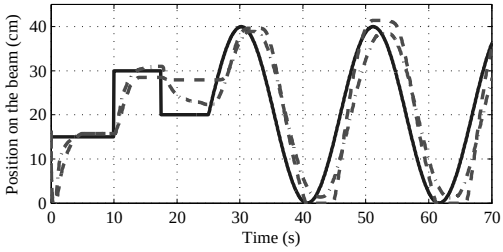


Figure 6: Neuro-fuzzy controller of the ANFIS model (dashed), reference (solid) and PID (dashdot).

immobile. The PID controller has a control signal with many more oscillations than does ANFIS and, thus, manages to avoid interference from dry attrition. When the reference received a sine signal, the neuro-fuzzy controller follows the reference as well as the PID controller and sometimes more accurately. However, at low velocities (extremes above and below the graphs), accuracy is reduced by dry attrition.

3.2 Results with NEFCON

For this model two inputs with five triangular functions each were used, and one output, the angle of reference, with seven singleton-type MFs. The input MFs are initiated with 50% overlap and symmetrically divided within their actuation interval, which is supplied by the designer. The maximum number of active rules for the system is 25. The learning constant for the controller was considered to be 10^{-3} and the sampling period for this model was 0.02 seconds. This value is due to the fact that the NEFCON model has a lower computational cost than that of the ANFIS model.

The position error of the ball and its variation were defined as system inputs. The value desired for both inputs was defined as zero and the execution range used for these variables was between -20 and 20 cm for the error and between -0.6 and 0.6 cm/s for the error variation. With respect to the output variable, the reference angle can vary from -45 to 45 degrees. The initial rule base was considered empty, except for the central rule, which received the output value equivalent to the zero reference angle.

As previously mentioned, NEFCON has its learning phase in real time. To show the development of this learning while the algorithm attempts to control the plant, a sine wave in the reference beam was used as a learning sequence. The rule creation phase was the first to enter into operation and its result is presented in table 1, where NH = Negative high, NL = Negative low, ZE = Zero, PL = Positive low, PH = Positive high, NE = Negative and PO = Positive.

Table 1: Rules found after 35 seconds of rule creation.

		Error variation				
		NH	NL	ZE	PL	PH
E	NH	-	-	-	-	-
R	NL	NH	-	NE	PO	PH
R	ZE	NH	NL	ZE	PL	-
O	PL	NH	NL	PL	PH	PH
R	PH	-	-	-	-	-

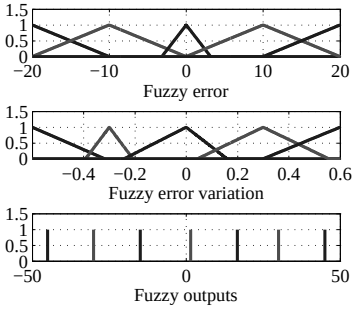


Figure 7: Membership functions after NEFCON adjust.

The second part of the algorithm functions as a fine adjustment of the fuzzy parameters by only moving the MFs while searching for an optimal adjustment. The MFs have the shape illustrated in fig. 7.

Observe that a situation of non symmetry occurred in the central MF (corresponding to membership ZE) of the second input (Error variation) of the neuro-fuzzy controller. Even though the algorithm tends to adjust the two sides symmetrically, the restrictions to gap formation between the MFs enable this asymmetric condition. We can observe also the compensation supplied by the algorithm for the output MF to calibrate the zero angle of the plant.

The result obtained with the reference sequence is shown in figure 8.

A numerical comparison between the results obtained for this second reference sequence is presented in table 2.

This table, despite showing the superiority of the

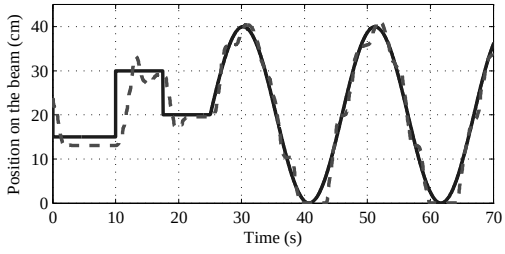


Figure 8: Neuro-fuzzy controller of the NEFCON model (dashed) and reference (solid).

Table 2: Numerical comparison between the controllers.

METHOD	NEFCON	ANFIS	PID
IAE	2.13	5.55	5.29
ISE	10.82	49.12	41.94

controller that used the NEFCON model, does not serve as a comparison parameter, since ANFIS copied the PID and NEFCON is optimal by its own structure.

4 CONCLUSIONS

This study showed how to implement and apply two neuro-fuzzy techniques to generate controllers (ANFIS and NEFCON). A number of applications were developed and discussed in such a way that both techniques could be applied successfully.

The controller based on the ANFIS model was developed with learning based on the backpropagation algorithm, using training pairs extracted from a PID controller and, in addition to controlling the ball and beam system, eliminated the high control signal variations in the learning stage. This fact makes ANFIS more susceptible to the effects of dry attrition present in the system. The controller based on the NEFCON model used a learning technique based on reinforcement learning with a number of alterations, and also achieved satisfactory results, since it successfully implemented the task of controlling the ball and beam system.

When compared to ANFIS, NEFCON is less complex to use, requiring it only to inform the algorithm on the admissible intervals for the input and output variables of the controller. This greater facility enables learning that is more directed toward user needs.

Optimization techniques such as genetic algorithms and ant colonies, among others, are some other techniques that may promote an even greater improvement in the hybrid models and might include future characteristics such as the mutation or evolution of the various forms associated to the structures. Another factor that could be analyzed is the compatibility of the models developed, in order to apply both in a same plant, in such a way that these models would collaborate or agree with each other.

ACKNOWLEDGEMENTS

The authors would like to thank CAPES-Brasil and CENPES-Petrobras for the financial support.

REFERENCES

Coelho, L. S. and Coelho, A. A. R. (1999). Algoritmos evolutivos em identificao e controle de processos: uma viso integrada e perspectivas. *SBA Controle & Automao*, 10(1):13–30.

Haykin, S. (2001). *Redes Neurais Artificiais: Princpios e prtica*. Bookman, 2 edition.

Jang, J. R. and S., C.-T. (1995). Neuro-fuzzy modeling and control. *Proceedings of the IEEE*, 83(3):378–406.

Jang, J.-S. R., Sun, C.-T., and Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing, A Computational Approach to Learn and Machine Intelligence*. Prentice-Hall, 1 edition.

Nrnberger, A., Nauck, D., and Kruse, R. (1999). Neuro-fuzzy control based on the nefcon-model: Recent developments. *Soft Computing 2*, pages 168–182.

Oliveira, J. P. B. M. (1994). Review of auto-tuning techniques for industrial pi controllers. Master's thesis, University of Salford.

Qiang, S., Zhou, Q., Gao, X. Z., and Yu, S. (2008). Anfis controller for double inverted pendulum. *The IEEE International Conference on Industrial Informatics*, pages 475–480.

Rezende, J. A. D. and Maitelli, A. L. (1999). Um esquema de neurocontrole com treinamento em tempo real aplicado ao posicionamento de um servomotor. *Simpsio Brasileiro de Automao Inteligente*.

Rodrigues, M. C., de Arajo, F. M. U., and Maitelli, A. L. (2006). Controladores neuro-fuzzy para sistemas nolineares. *Congresso Brasileiro de Automtica*.

Rodrigues, M. C., Maitelli, A. L., and de Arajo, F. M. U. (2004). Controle neuro-fuzzy com treinamento em tempo real aplicado a um sistema ball and beam. *Congresso Brasileiro de Automtica*.

Shujaec, K., Sarathy, S., Nicholson, R., and George, R. (2002). Neuro-fuzzy controller and convention controller: a comparison. *World Automation Congress*, 13:207–213.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning, An Introduction*. Bradford Book, 1 edition.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8:338–353.

IFT-BASED PI-FUZZY CONTROLLERS
Signal Processing and Implementation

Radu-Emil Precup, Mircea-Bogdan Rădac, Stefan Preitl
Dept. of Automation and Appl. Inf., "Politehnica" University of Timisoara
Bd. V. Parvan 2, 300223 Timisoara, Romania
radu.precup@aut.upt.ro, mircea.radac@aut.upt.ro, stefan.preitl@aut.upt.ro

Marius-Lucian Tomescu
Fac. of Computer Sci., "Aurel Vlaicu" University of Arad, Complex Univ. M
Str. Elena Dragoi 2, 310330 Arad, Romania
tom_uav@yahoo.com

Emil M. Petriu
School of Information Technology and Eng., University of Ottawa
800 King Edward, Ottawa, ON, K1N 6N5 Canada
petriu@site.uottawa.ca

Adrian Sebastian Paul
Dept. of Automation and Appl. Inf., "Politehnica" University of Timisoara
Bd. V. Parvan 2, 300223 Timisoara, Romania
adi11p@yahoo.com

Keywords: Iterative Feedback Tuning, PI-fuzzy controllers, Stability analysis.

Abstract: New Takagi-Sugeno PI-fuzzy controllers (PI-FCs) are suggested in this paper. The PI-FC design is based on the optimization of PI controllers in terms of the Iterative Feedback Tuning (IFT) approach. Next the parameters of the PI controllers are mapped onto the parameters of the Takagi-Sugeno PI-FCs in terms of the modal equivalence principle. An attractive design method is derived to support the implementation of low-cost PI-FCs. The design is enabled by a stability analysis theorem based on Lyapunov's theorem for time-varying systems. The theoretical approaches are validated by a case study corresponding to the position control of a servo system. Real-time experimental results are included.

1 INTRODUCTION

The design of control systems (CSs) making use of measurement data is successful in many industrial applications without models available for the controlled process. The time-consuming design of those models can be avoided. Fuzzy control is an alternative when very good steady-state and dynamic CS performance indices can be guaranteed. The systematic design of fuzzy controllers must be assisted by the analysis of the fuzzy CS structural properties i.e. stability, controllability, parametric sensitivity and robustness (Sala et al., 2005; Kovačić and Bogdan, 2006; Blažič and Škrjanc, 2007).

Iterative Feedback Tuning (IFT) is a gradient-based approach, based on input-output data recorded from the closed-loop system (Hjalmarsson et al., 1998). The performance specifications are expressed in terms of objective functions in appropriate optimization problems. Those problems can be solved by iterative gradient-based minimization implemented as IFT algorithms. IFT makes use of closed-loop experimental data to calculate the estimates of the gradients of the objective functions. Several experiments are done per iteration and the updated controller parameters are calculated based on the input-output data. So the IFT belongs to the direct data-based offline-adaptive controller designs.

The combination of IFT and fuzzy control leads

to the convenient performance enhancement of fuzzy CSs after their initial tuning (Precup et al., 2008). The first contribution of the paper concerns the modification of the second experiment specific to IFT to be overlapped over the normal CS operation. Several useful remarks are introduced in relation with the signal processing and implementation of the IFT algorithms. The second contribution is a new design method of low-cost Takagi-Sugeno PI-FCs which is based on mapping the results from the linear case onto the fuzzy one in terms of the modal equivalence principle (Galichet and Foulloy, 1995). The third contribution is a stability analysis theorem based on Lyapunov's theorem for time-varying systems derived from (Slotine and Li, 1991) to support the PI-FC design.

The paper is organized as follows. Section 2 discusses the signal processing and implementation aspects regarding the IFT algorithm. Next, Section 3 presents the new design method for a class of Takagi-Sugeno PI-FCs. Section 4 addresses a case study associated with low-cost implementations of DC drive servo system position CSs. The conclusions are presented in Section 5.

2 SIGNAL PROCESSING AND IMPLEMENTATION ASPECTS IN IFT ALGORITHMS

The IFT-based CS structure is presented in Figure 1, where: r – the reference input, d – the disturbance input, $e = r - y$ – the control error, u – the control signal, $\boldsymbol{\rho}$ – the parameters vector having the controller tuning parameters as its components, $C(\boldsymbol{\rho})$ – the transfer function of the linear controller, a PI one, here to be replaced by the PI-FC, in order to improve the CS performance indices, F – the transfer function of the reference model that prescribes the desired behaviour to be exhibited by the closed-loop system, P – the transfer function of the controlled process, y – the controlled output, y_d – the desired output produced by the reference model, $\delta y = y - y_d$ – the model tracking error, and IFT – the Iterative Feedback Tuning algorithm, $\mathbf{i}$ – the input vector to set the performance specifications.

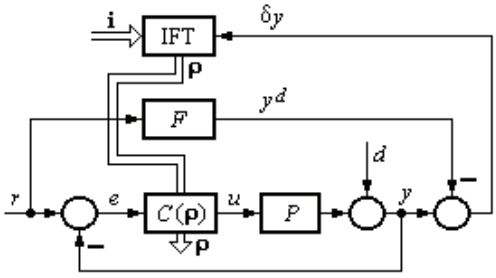


Figure 1: CS structure with IFT.

The operational variable in the transfer functions has been omitted for the sake of simplicity. However that variable will be mentioned in the sequel in the well accepted notation s for continuous-time systems and z for discrete-time ones to improve the clarity of the presentation when needed. That is also the reason for inserting or removing the argument $\boldsymbol{\rho}$.

The controller parameterization is such that the transfer function $C(\boldsymbol{\rho})$ is differentiable with respect to $\boldsymbol{\rho}$. The controller must ensure an initially stabilized CS. The initial controller tuning affects the convergence of the iterative process.

The accepted expression of the objective function $J(\boldsymbol{\rho})$ is

$$J(\boldsymbol{\rho}) = (0.5/N)\sum_{k=1}^{N}[\delta y(k,\boldsymbol{\rho})]^2, \tag{1}$$

where: N – the number of samples setting the length of each experiment. A typical objective is to find a parameters vector $\boldsymbol{\rho}^*$ to minimize $J(\boldsymbol{\rho})$ and make the error δy tend to zero by the optimization problem

$$\boldsymbol{\rho}^* = \arg\min_{\boldsymbol{\rho}\in SD} J(\boldsymbol{\rho}), \tag{2}$$

where several constraints can be imposed regarding the controlled process or the closed-loop CS. The most important constraint accepted in this paper concerns the necessity of stable CSs, and SD stands for the stability domain. Other variables including the control signal can be used. That requires additional signal processing and increased cost.

The IFT algorithms solve the optimization problem (2) by means of numerical optimization techniques. Newton's method is popular in IFT since it can be treated independently of the difficulties inherent to the model-based techniques. It evaluates repeatedly a new solution based on a point of the function and its approximate derivative. The mathematical formulation is the following update law to calculate the next set of parameters $\boldsymbol{\rho}^{i+1}$:

$$\boldsymbol{\rho}^{i+1} = \boldsymbol{\rho}^{i} - \gamma^{i}(\mathbf{R}^{i})^{-1} est[\frac{\partial J}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}^{i})], \tag{3}$$

where: i – the index of current iteration, $est[\frac{\partial J}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}^{i})]$ – the estimate of the gradient vector, γ^{i} – the step size, $\boldsymbol{\rho}^{0}$ – the initial guess of the tuning parameters, and $\mathbf{R}^{i}$ – a regular positive definite matrix. $\mathbf{R}^{i}$ can be the Hessian or the identity matrix to simplify the signal processing.

Differentiating (1), the gradient becomes

$$\frac{\partial J}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}^{i}) = (1/N)\sum_{k=1}^{N}[\delta y(k,\boldsymbol{\rho}^{i})\frac{\partial \delta y}{\partial \boldsymbol{\rho}}(k,\boldsymbol{\rho}^{i})]\cdot \tag{4}$$

To calculate the general expressions of the gradient of the output error it is necessary to make use of the information obtained from the closed-loop system. The sensitivity function S and the complementary sensitivity function T must be expressed:

$$\begin{aligned} &S(\boldsymbol{\rho}) = 1/[1 + C(\boldsymbol{\rho})P], \\ &T(\boldsymbol{\rho}) = 1 - S(\boldsymbol{\rho}) = C(\boldsymbol{\rho})P/[1 + C(\boldsymbol{\rho})P]. \end{aligned} \tag{5}$$

The differentiation of $\delta y(\boldsymbol{\rho})$ making use of (5) and Figure 1 leads to the gradient of $\delta y(\boldsymbol{\rho})$:

$$\frac{\partial \delta y}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}) = \frac{1}{C(\boldsymbol{\rho})}\cdot\frac{\partial C}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho})T(\boldsymbol{\rho})(r - y)\cdot \tag{6}$$

To obtain the estimate of the gradient of the model tracking error use is made of two experiments per iteration for the PI controllers. In the first experiment, the normal one, use is made of Figure 1, the reference input r_1 is applied to the CS and the controlled output y_1 is measured. In the second experiment, the gradient one, the control error of the first experiment $e_1 = r_1 - y_1$ is applied as the reference input r_2 (Hjalmarsson et al., 1998) and the controlled output y_2 is measured. That processing is far away from the normal CS operation. Therefore a new gradient experiment is suggested here where the reference input r_2 is applied and the signal e_1 is injected after the control signal. That can be expressed as the experimental scheme for the gradient experiment illustrated in Figure 2 where the blocks F and IFT have been dropped out.

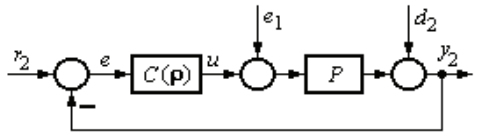


Figure 2: Experimental scheme for gradient experiment.

Accepting the lower subscript pointing out the index of the current experiment, the reference input and controlled output in the normal experiment are

$$r_1 = r,\ y_1(\boldsymbol{\rho}) = T(\boldsymbol{\rho})r + S(\boldsymbol{\rho})d_1. \tag{7}$$

For the gradient experiment, making use of the Figure 2, the results are

$$\begin{aligned} &r_2 = K\,r,\ y_2(\boldsymbol{\rho}) = T(\boldsymbol{\rho})r_2 + \\ &+ \{P/[1 + C(\boldsymbol{\rho})P]\}(r_1 - y_1(\boldsymbol{\rho})] + S(\boldsymbol{\rho})d_2, \end{aligned} \tag{8}$$

where the gain K has been inserted to show the proportional reference inputs in the two experiments. Next (7) is multiplied by K, extracted from (8), the relationship (6) is used and the result becomes

$$\begin{aligned} &\frac{\partial \delta y}{\partial \boldsymbol{\rho}}(k,\boldsymbol{\rho}^{i}) = \frac{\partial C}{\partial \boldsymbol{\rho}}(q^{-1},\boldsymbol{\rho}^{i})[y_2(k,\boldsymbol{\rho}^{i}) - \\ &- K\,y_1(k,\boldsymbol{\rho}^{i})] - S(\boldsymbol{\rho}^{i})[d_2(k) - K\,d_1(k)]/ \\ &C(q^{-1},\boldsymbol{\rho}^{i}). \end{aligned} \tag{9}$$

The second term in the right-hand side of (9) depends on the disturbance inputs, it affects the gradient, so it should be alleviated. Neglecting that term the estimate of the gradient of δy is evaluated:

$$\begin{aligned} &est[\frac{\partial \delta y}{\partial \boldsymbol{\rho}}(k,\boldsymbol{\rho}^{i})] = \frac{\partial C}{\partial \boldsymbol{\rho}}(q^{-1},\boldsymbol{\rho}^{i})[y_2(k,\boldsymbol{\rho}^{i}) - \\ &- K\,y_1(k,\boldsymbol{\rho}^{i})]. \end{aligned} \tag{10}$$

The alleviation of the second term in the right-hand side of (9) can be done by the proper initial tuning of the controller parameters because $S(\boldsymbol{\rho}^{i})$ plays the role of filter. That term can lead to shifted estimates with negative effects on the convergence. A similar approach (Hildebrand et al., 2005) is characterized by an additional prefilter designed as solution to optimization problems. That filter is not introduced here to simplify the signal processing accepting that K=1. The role of $\boldsymbol{\rho}^{0}$ is highlighted from that point of view.

Summarizing all signal processing aspects mentioned before in the linear case, one iteration in the IFT algorithm consists of the following steps.

Step 0. Set $\boldsymbol{\rho}^{0}$.

Step 1. The two experiments are done making use of the CS structures presented in Figure 1 and Figure 2 and the outputs y_1 and y_2 are measured.

Step 2. The output of the reference model is generated, y_d, and the output error δy is calculated.

Step 3. The estimate of gradient of J is calculated according to (4) and (10).

Step 4. The next set of parameters $\boldsymbol{\rho}^{i+1}$ is calculated in terms of the update law (3).

3 DESIGN OF TAKAGI-SUGENO PI-FUZZY CONTROLLERS

The Takagi-Sugeno PI-FC is a discrete-time controller built around the two inputs-single output fuzzy controller (TISO-FC), Figure 3, where $\Delta e(k)=e(k)-e(k-1)$ and $\Delta u(k)=u(k)-u(k-1)$ is the increment of control error and signal, respectively. The fuzzification is done by the membership functions presented in Figure 4, the inference engine employs the MAX and MIN operators assisted by the rule base presented in Table 1, and the weighted average defuzzification method is employed.

Figure 3: Structure of Takagi-Sugeno PI-fuzzy controller.

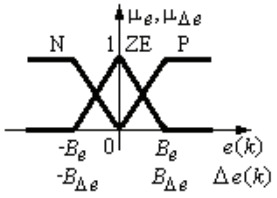


Figure 4: Input membership functions of TISO-FC.

Table 1: Rule base as decision table of TISO-FC.

$\Delta e(k)$	$e(k)$		
	N	ZE	P
P	$\Delta u_k = f_k$	$\Delta u_k = f_k$	$\Delta u_k = \eta f_k$
ZE	$\Delta u_k = f_k$	$\Delta u_k = f_k$	$\Delta u_k = f_k$
N	$\Delta u_k = \eta f_k$	$\Delta u_k = f_k$	$\Delta u_k = f_k$

The rule base of the PI-FC can be reduced to two rules (Johanyák and Kovács, 2007). The rule consequents (Table 1) point out the term $f(k)$:

$$f(k)= K_P[\Delta e(k)+\alpha e(k)]. \quad (11)$$

Eq. (11) corresponds to the recurrent equation of an incremental digital PI controller. The Takagi-Sugeno PI-FCs will exhibit as bumpless interpolators between two linear PI controllers. The additional parameter η with typical values within $0<\eta<1$ reduces the overshoot.

The parameters K_P and α in (11) can be obtained either directly in the discrete-time form or by the continuous-time form of the PI controller

$$C(s) = k_c(1+sT_i)/s = k_C[1+1/(sT_i)], \quad (12)$$

followed by the discretization in terms of the sampling period T_s (in terms of quasi-continuous control), where T_i is the integral time constant and k_C, $k_C = T_i k_c$, is the controller gain. In case of Tustin's discretization method applied here the parameters K_P and α obtain the expressions

$$K_P = k_C[1-T_s/(2T_i)], \alpha = 2T_s/(2T_i - T_s). \quad (13)$$

Accepting the approximations specific to the quasi-continuous digital control (Precup et al., 2008) the Takagi-Sugeno PI-FCs can be considered as continuous-time fuzzy controllers. However the calculation of the maximum T_s such that the stability is also ensured is of interest. The IFT-based design method dedicated to the accepted class of Takagi-Sugeno PI-FCs consists of the following steps.

Step 1. T_s is set and an initial linear tuning method is applied to calculate the initial controller parameters, K_P and α. They can be obtained also by an initial guess based on the designer's experience.

Step 2. The initial data of the IFT algorithm and the reference model parameters are set.

Step 3. The IFT algorithm presented in the previous Section is applied resulting in the optimal controller parameters.

Step 4. B_e and η are chosen according to the performance specifications. The stability analysis to be presented as follows is taken into account. Next the modal equivalence principle is applied:

$$B_{\Delta e} = \alpha B_e. \quad (14)$$

The current trends in the stability analysis of fuzzy CSs employ Lyapunov's (Wang et al., 2007), Krasovskii's and La Salle's approaches (Tian and Peng, 2006), the describing function method or algebraic approaches (Michels et al., 2006; Jantzen, 2007). The stability analysis to be presented as follows employs the formalism applied in (Lam and Leung, 2008; Lam and Ling, 2008). The state-space equation of the controlled process is

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x},t) + \mathbf{b}(\mathbf{x},t)u(t), \mathbf{x}(t_0) = \mathbf{x}_0, \quad (15)$$

where $\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_n]^T \in D$ is the state vector, $n \in N^*$, $\dot{\mathbf{x}} = [\dot{x}_1 \quad \dot{x}_2 \quad \dots \quad \dot{x}_n]^T$ is the derivative of $\mathbf{x}$ with respect to the independent time variable t, $\mathbf{f}, \mathbf{b} : D \times [0,\infty) \to R^n$ are continuous functions of t:

$$\mathbf{f}(\mathbf{x},t) = [f_1(\mathbf{x},t) \quad f_2(\mathbf{x},t) \quad \dots \quad f_n(\mathbf{x},t)]^T, \quad (16)$$
$$\mathbf{b}(\mathbf{x},t) = [b_1(\mathbf{x},t) \quad b_2(\mathbf{x},t) \quad \dots \quad b_n(\mathbf{x},t)]^T,$$

T stands for matrix transposition, and the disturbance is absent. The PI-FC inputs are (n=2):

$$x_1(t) = e(t) = r - y(t) = e(k),$$
$$x_2(t) = \dot{x}_1(t) = \Delta e(k) / T_s. \quad (17)$$

The expression of the control signal is

$$u = (\sum_{i=1}^{r_B} \alpha_i u_i) / (\sum_{i=1}^{r_B} \alpha_i), \quad (18)$$

where u_i is the control signal produced in the consequent of the *i*-th rule, $i = \overline{1, r_B}$, r_B is the number of fuzzy control rules, and α_i is the firing strength (Precup et al., 2008).

The Lyapunov function candidate is

$$V : D \times [0, \infty) \to R, V(\mathbf{x}, t) = g(t)\mathbf{x}^T \mathbf{P}\, \mathbf{x}, \quad (19)$$

where $\mathbf{P} \in R^{n \times n}$ is a constant positive definite matrix and $g : [0, \infty) \to [0, \infty)$ is a continuously differentiable function. The derivative of V with respect to time with the system constrained to (15) is

$$\dot{V}(\mathbf{x}, t) = F(\mathbf{x}, t) + B(\mathbf{x}, t)u, F(\mathbf{x}, t) = \quad (20)$$
$$= g(t)[\mathbf{f}(\mathbf{x}, t)]^T \mathbf{P}\, \mathbf{x} + g(t)\mathbf{x}^T \mathbf{P}\, \mathbf{f}(\mathbf{x}, t) +$$
$$+ \dot{g}(t)\mathbf{x}^T \mathbf{P}\, \mathbf{x}, B(\mathbf{x}, t) = g(t)[\mathbf{b}(\mathbf{x}, t)]^T \mathbf{P}\, \mathbf{x} +$$
$$+ g(t)\mathbf{x}^T \mathbf{P}\, \mathbf{b}(\mathbf{x}, t),$$

and its expression calculated for $u = u_k(\mathbf{x})$ is $\dot{V}_k(\mathbf{x}, t)$.

Theorem 1. Let $\mathbf{x} = \mathbf{0} \in D \subset R^n$ be an equilibrium point for (15) controlled by the accepted PI-FC and V the Lyapunov function candidate (19) such that the following two conditions are fulfilled:

$$V(\mathbf{x}, t) \le W^1(\mathbf{x}), \quad (21)$$
$$\dot{V}_k(\mathbf{x}, t) \le -W_k^2(\mathbf{x}), k = \overline{1, r_B}, \forall t \ge 0, \forall \mathbf{x} \in D,$$

where W^1 and W_k^2 are continuous positive definite functions on D. Then $\mathbf{x} = \mathbf{0}$ will be uniformly asymptotically stable.

Proof: Use is made of (20) and (21) leading to

$$\dot{V}(\mathbf{x}, t) \le -[\sum_{k=1}^{r_B} (W_k^2(\mathbf{x})\alpha_k(\mathbf{x}))] / [\sum_{k=1}^{r_B} \alpha_k(\mathbf{x})]. \quad (22)$$

Therefore Lyapunov's theorem for time-varying systems is fulfilled due to the conditions (21) and (22), and the equilibrium point $\mathbf{x} = \mathbf{0}$ will be uniformly asymptotically stable.

Theorem 1 offers sufficient stability conditions in the choice of the parameters B_e and η. Its application has been implemented for a real-world process in the Intelligent Systems Laboratory with the "Politehnica" University of Timisoara (PUT).

4 CASE STUDY

The experimental setup is built around the INTECO DC servo system with backlash laboratory equipment, Figure 5, with rated amplitude of 24 V, current of 3.1 A, torque of 15 N cm and speed of 3000 rpm. The inertial load weighs 2.03 kg.

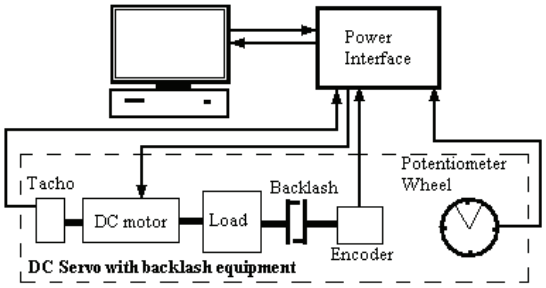


Figure 5: Structure of experimental setup.

The transfer functions of the simplified controlled process and reference model are

$$P(s) = \frac{k_P}{s(1 + sT_\Sigma)}, \; F(s) = \frac{1}{s^2 + 1.5s + 1}, \quad (23)$$

$k_P = 139.88$, $T_\Sigma = 0.9198\,\text{s}$. The continuous-time PI controller has been obtained by frequency domain design which yields the parameters $k_C = 0.01036$ and $T_i = 3.1043\,\text{s}$. Discretizing with $T_s = 0.01\,\text{s}$, the initial digital PI controller parameters are $\boldsymbol{\rho}^0 = [K_P = 0.01035 \quad \alpha = 0.0029]^T$. The parameters obtained after 10 iterations for the step size $\gamma^i = 10^{-6}$ are $\boldsymbol{\rho}^{10} = [0.010346 \quad 0.003226]^T$. Setting $B_e = 20$ and $\eta = 0.5$, (14) results in $B_{\Delta e} = 0.06452$.

The constant reference input $r = 40\,\text{rad}$ has been applied. The behaviour of the CS with linear controller before IFT is illustrated in Figure 6.

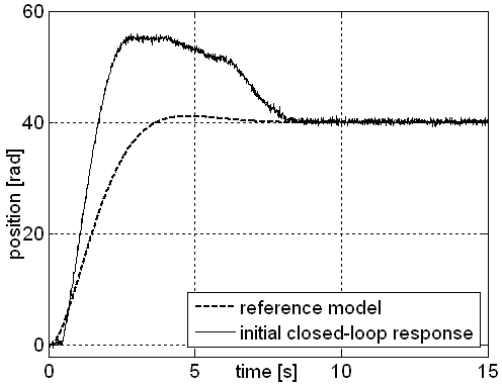


Figure 6: Reference model output and controlled output (position) versus time for linear CS before IFT.

The behaviour of the CS with PI-FC after IFT is presented in Figure 7. The performance indices (overshoot and settling time) of the CS have been improved. A band-limited white noise of variance 0.01 has been applied as the disturbance input *d*.

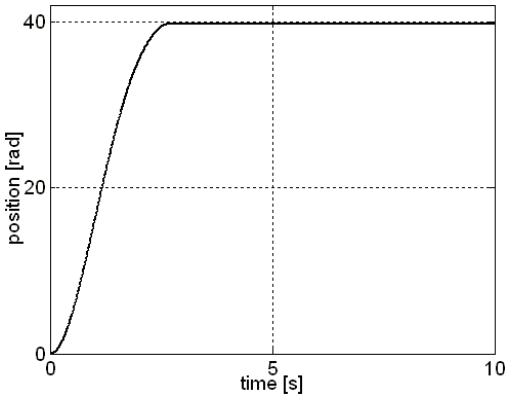


Figure 7: *y* versus time for fuzzy CS after IFT.

5 CONCLUSIONS

The paper has proposed a stable design method dedicated to a class of Takagi-Sugeno PI-FCs. It is based on mapping the IFT-based linear case results onto the fuzzy control results.

Several signal processing aspects regarding the simplification of the implementation have been discussed. They involve an original gradient experiment. A single gradient experiment is needed. However an additional one can be employed in other CS structures.

The stability analysis can be applied to the fuzzy control of time-varying systems. It is valid because of the quasi-continuous digital implementation of the controllers that enables the controller design.

One future research topic concerns the convergence analysis. Although the stability analysis suggested is attractive, the convergence is not guaranteed. The future research will be dedicated to the application of the approaches to other fuzzy controller structures (Valente de Oliveira and Gomide, 2001; Vaščák, 2007; Pedrycz, 2009).

ACKNOWLEDGEMENTS

The paper was supported by the CNMP & CNCSIS of Romania. The second and sixth authors are doctoral students with the PUT. The second author is an SOP HRD stipendiary co-financed by the European Social Fund through the project ID 6998.

REFERENCES

Blažič, S., Škrjanc, I., 2007. Design and stability analysis of fuzzy model-based predictive control - a case study. *J. of Intelligent and Robotic Systems*. 49, 279-292.

Galichet, S., Foulloy, L., 1995. Fuzzy controllers: synthesis and equivalences. *IEEE Transactions on Fuzzy Systems*. 3, 140-148.

Hildebrand, R., Lecchini, A., Solari, G., Gevers, M., 2005. Optimal prefiltering in Iterative Feedback Tuning. *IEEE Trans. Autom. Control*. 50, 1196-1200.

Hjalmarsson, H., Gevers, M., Gunnarsson, S., Lequin, O., 1998. Iterative Feedback Tuning: theory and applications. *IEEE Control Syst. Magazine*. 18, 26-41.

Jantzen, J., 2007. *Foundations of Fuzzy Control*. Chichester: Wiley.

Johanyák, Z. C., Kovács, S., 2007. Sparse fuzzy system generation by rule base extension. In *Proc. 11th International Conference on Intelligent Engineering Systems (INES 2007)*. Budapest, Hungary, 99-104.

Kovačić, Z., Bogdan, S., 2006. *Fuzzy Controller Design: Theory and Applications*. Boca Raton, FL: CRC Press.

Lam, H. K., Leung, F. H. F., 2008. Stability analysis of discrete-time fuzzy-model-based control systems with time delay: time-delay independent approach. *Fuzzy Sets and Systems*. 159, 990-1000.

Lam, H. K., Ling, B. W. K., 2008. Sampled-data fuzzy controller for continuous nonlinear systems. *IET Control Theory & Applications*. 2, 32-39.

Michels, K., Klawonn, F., Kruse, R., Nürnberger, A., 2006. *Fuzzy Control: Fundamentals, Stability and Design of Fuzzy Controllers*. Berlin, Heidelberg, New York: Springer Verlag.

Pedrycz, W., 2009. From fuzzy sets to shadowed sets: interpretation and computing. *International Journal of Intelligent Systems*. 24, 48-61.

Precup, R.-E., Preitl, S., Rudas, I. J., Tomescu, M. L., Tar, J. K., 2008. Design and experiments for a class of fuzzy controlled servo systems. *IEEE/ASME Transactions on Mechatronics*. 13, 22-35.

Sala, A., Guerra, T. M., Babuška, R., 2005. Perspectives of fuzzy systems and control. *Fuzzy Sets and Systems*. 156, 432-444.

Slotine, J.-J., Li, W., 1991. *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice-Hall.

Tian, E., Peng, C., 2006. Delay-dependent stability analysis and synthesis of uncertain T-S fuzzy systems with time-varying delay. *Fuzzy Sets and Systems*. 157, 544-559.

Valente de Oliveira, J., Gomide, F. A. C., 2001. Formal methods for fuzzy modeling and control. *Fuzzy Sets and Systems*. 121, 1-2.

Vaščák, J, 2007. Navigation of mobile robots using potential fields and computational intelligence means. *Acta Polytechnica Hungarica*. 4, 63-74.

Wang, W.-J., Chen, Y.-J., Sun, C.-H., 2007. Relaxed stabilization criteria for discrete-time T-S fuzzy control systems based on a switching fuzzy model and piecewise Lyapunov function. *IEEE Trans. Systems, Man, and Cyber., Part B: Cybernetics*. 37, 551-559.

TRANSITION VELOCITY FUNCTION FOR IMPULSE CONTROL SYSTEMS

Stephen van Duin[1], Matthias Ahlswede[2] and Christopher D. Cook[1]
[1] *Faculty of Engineering, University of Wollongong, Northfields Avenue, Gwynnville, Australia*
{svanduin, ccook}@uow.edu.au
[2] *Institute of Production Engineering and Machine Tools, Leibniz University Hannover, Hannover, Germany*
m.ahlswede@gmx.net

Keywords: Impulsive Control, Static Friction, Limit Cycle, Stick-slip, Impulse Shape, Friction Model, Accuracy.

Abstract: This paper presents a modified impulse controller that is used to improve the velocity tracking of a servomechanism having characteristics of high nonlinear friction. A hybrid control scheme consisting of a conventional PID part and an impulsive part is used as a basis to the modified controller. This has previously been used to improve the position and velocity tracking of robot manipulators at very low velocities. Experiments show that at higher velocities the improved performance of the impulse part of the hybrid controller diminishes and can be counterproductive at these speeds when compared to conventional PID control alone. The modified hybrid impulse controller in this paper uses a mathematical function to transition the amount of torque from an impulse as a function of velocity to achieve more precise tracking across a range of velocities.

1 INTRODUCTION

Precision robot manufacturers continually strive to increase the accuracy of their machinery in order to remain competitive. The ability of a robot manipulator to position its tool centre point to within a very high accuracy allows the robot to be used for more precise tasks. For positioning of a tool centre point, the mechanical axes of a robot will be required to be precisely controlled around zero velocity where friction is highly non-linear and difficult to control. Furthermore, precise velocity control at high velocities is typically required for increased productivity. Each axis of a robot is typically controlled by a servomechanism and this paper deals with improving the control of these basic robot components in the presence of friction.

Nonlinear friction is inherently present in all mechanisms and can cause stick-slip during precise positioning. In many instances, stick-slip has been reduced or avoided by modifying the mechanical properties of the system; however this approach may not always be practical or cost effective. Alternatively, advances in digital technology have made it possible for the power electronics of servomechanisms to be controlled with much greater flexibility. By developing better controllers, the unfavourable effects of non-linear friction may be reduced or eliminated completely.

Impulse control has been successfully used for accurate positioning of servomechanisms with high friction where conventional control schemes alone have difficulty in approaching zero steady state error. Static and Coulomb friction can cause a conventional PID controller having integral action (I), to overshoot and limit cycle around the reference position. This is particularly a problem near zero velocities where friction is highly non linear and the servomechanism is most likely to stick-slip. Despite the above difficulties, PID controllers are still widely used in manufacturing industries because of their relative simplicity and reasonable robustness to parameter uncertainty and unknown disturbances.

Stick-slip can be reduced or eliminated by using impulsive control near or at zero velocities. The impulsive controller is used to overcome static friction by impacting the mechanism and moving it by microscopic amounts. By combining the impulsive controller and conventional controller the PID part can be used to provide stability. Moving towards the reference position the impulse controller is used to improve accuracy for the final positioning where the error signal is small.

By applying a short impulse of sufficient force, plastic deformation occurs between the asperities of mating surfaces resulting in permanent controlled movement. If the initial pulse causes insufficient movement, the impulsive controller produces additional pulses until the position error is reduced to a minimum.

A number of investigators have devised impulsive controllers which achieve precise motion in the presence of friction by controlling the height or width of a pulse. Yang and Tomizuka (Yang et al., 1988) applied a standard rectangular shaped pulse in which the height of the pulse was a force 3 to 4 times greater than the static friction to guarantee movement. The width of the pulse was adaptively adjusted proportional to the error and was used to control the amount of energy required to move the mechanism towards the reference position. Alternatively, Popovic (Popovic et al., 2000) described a fuzzy logic pulse controller that determined both the optimum pulse amplitude and pulse width simultaneously using a set of membership functions. Hojjat and Higuchi (Hojjat et al., 1991) limited the pulse width to a fixed duration of 1ms and varied the amplitude by applying a force about 10 times the static friction.

In a survey of friction controllers by Armstrong-Hélouvry (Armstrong- Hélouvry et al., 1994), it is commented that underlying the functioning of these impulsive controllers is the requirement for the mechanism to be in the stuck or stationary position before subsequent impulses are applied. Thus, previous impulse controllers required each small impacting pulse to be followed by an open loop slide ending in a complete stop.

van Duin (van Duin et al., 2006), used a hybrid PID + Impulsive controller to improve the precision of a robot manipulator arm in the presence of static and Coulomb friction. The design and functioning of the controller does not require the mechanism to come to rest between subsequent pulses, making it suitable for both point-to-point positioning and speed regulation. van Duin (van Duin et al., 2006) manipulated the pulse shape to match the dynamic friction by making this shape responsive to very small changes in velocity.

The error in positioning during different tracking tasks at zero and low velocities was greatly improved. However, further experiments showed that the PID + Impulse controller had greater errors at high velocities compared to a simple PID controller alone.

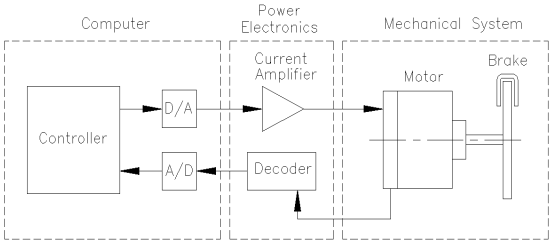


Figure 1: Experimental friction test bed.

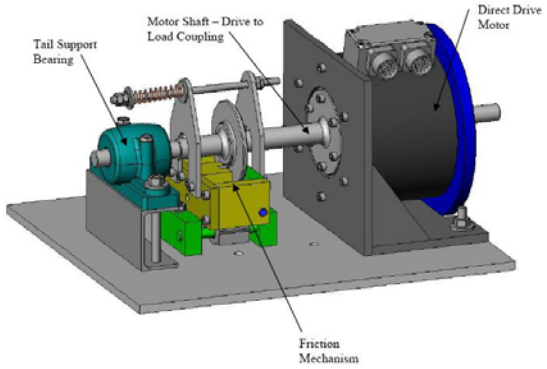


Figure 2: Three dimensional drawing of the friction test bed.

This paper presents a modified impulse controller where the impulsive part of the hybrid PID + Impulse controller is gradually disabled at higher velocities so that the conventional linear PID part is eventually solely providing the driving torque. It is shown that at greater velocities, the static and Coulomb friction is less influential and that the system is more dominated by the relatively linear viscous frictional effects. Using a transition function the performance is shown to be improved for both low and high velocities, while maintaining system stability.

2 EXPERIMENTAL SYSTEM

2.1 Servomechanism

For these experiments a purpose built single axis friction test-bed was used to simulate the conditions typically observed in an industrial robot arm.

Figures 1 and 2 show the experimental friction test bed system that consists of a single axis direct drive servomechanism actuator coupled to a friction generating disk brake. Torque is transmitted by the actuator to the friction mechanism through a direct coupled shaft to eliminate the presence of backlash, gear cogging, belt cogging etc. Direct drive isolates

the friction characteristics; however, the impulse control systems in this paper have been repeated on a Hirata ARi350 SCARA robot with comparable results.

Position data is obtained from a shaft encoder housed within the motor and has a maximum resolution of 2^{19} counts per revolution or 1.198e-5 rad/count. Digital torque control of the motor is achieved using a three phase direct drive servo amplifier.

Matlab's xPC target oriented server was used to provide control to the servomechanism drive. For these experiments the digital drive was used in current control mode. This means the output voltage from the 12-bit D/A converter gives a torque command to the actuator's power electronics, which has a time constant of 0.1ms.

The system controller was compiled and run using Matlab's real time xPC Simulink® block code.

2.2 Hybrid PID + Impulse Controller

Figure 3 shows the block diagram of a PID linear controller + impulsive controller. This hybrid controller has been suggested by Li (Li et al., 1998) where the PID driving torque and impulsive controller driving torque are summed together. It is unnecessary to stop at the end of each sampling period; therefore, the controller can be used for both position and speed control.

The controller can be divided into two parts; the upper part is the continuous driving force for large scale movement and control of external force disturbances. The lower part is an additional proportional controller k_{pwm} with a pulse width modulated sampled-data hold (PWMH), and is the basis of the impulsive controller for the control of stick-slip.

The system controller is sampled at 2 kHz. The impulse itself is sampled and applied at one twentieth of the overall sampling period (i.e. 100 Hz) to match the mechanical system dynamics. Figure 4 shows a typical output of the hybrid controller for one impulse sampling period τ_s. The pulse with a height f_p is added to the PID output. Because the PID controller is constantly active, the system has the ability to counteract random disturbances applied to the servomechanism. The continuous part of the controller is tuned to react to large errors and high velocity, while the impulse part is optimized for final positioning where stiction is most prevalent.

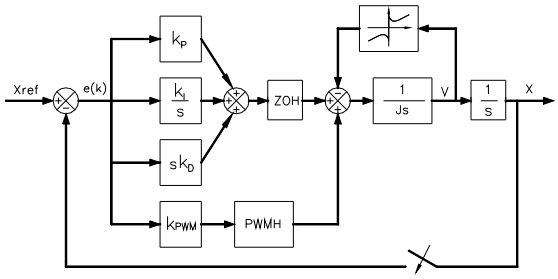


Figure 3: Block diagram of the experimental system controller.

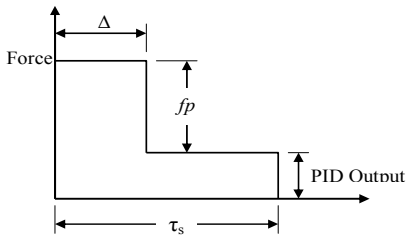


Figure 4: Friction controller output.

For large errors, the impulse width approaches the full sample period τ_s, and for very large errors, it transforms into a continuous driving torque. When this occurs, the combined control action of the PID controller and the impulsive controller will be continuous. Conversely, for small errors, the PID output is too small to have any substantial effect on the servomechanism dynamics.

The high impulse sampling rate, combined with a small error, ensures that the integral (I) part of the PID controller output has insufficient time to rise and produce limit cycling. To counteract this loss of driving torque, when the error is below a threshold, the impulsive controller begins to segment into individual pulses of varying width and becomes the primary driving force. One way of achieving this is to make the pulse width Δ determined by:

$$\Delta = \frac{k_{pwm} \cdot e(k)\tau_s}{f_p} \quad \text{if } \mathrm{k_{pwm}} \cdot |e(k)| \le |f_p| \qquad \Delta = \tau_s \quad \text{otherwise} \tag{1}$$

In (1)

$$f_p = \left|f_p\right| \cdot sign\left(e(k)\right) \tag{2}$$

where *e(k)* is the error input to the controller, $|f_p|$ is a fixed pulse height greater than the highest static

friction and τ_s is the overall sampling period.

For the experimental results described in this paper, the impulsive sampling period τ_s was 10ms and the pulse width could be incrementally varied by 1ms intervals. The pulse width gain k_{pwm}, is experimentally determined by matching the mechanism's observed displacement d to the calculated pulse width t_p using the equation of motion:

$$d = \frac{f_p(f_p - f_C)}{2mf_C} t_p^2, \quad f_p > 0 \qquad (3)$$

The gain is iteratively adjusted until the net displacement for each incremental pulse width is as small as practical.

To further improve the performance of the controller, van Duin (van Duin et al., 2006) use a modified impulse shape to better counteract the dynamics of friction. To overcome stiction, it is necessary to have an initial driving force greater than the static friction. Immediately after motion begins, the opposing friction reduces dramatically and, if motion continues, will be maintained at the Coulomb friction value. Figure 5 shows most of the effective energy of the pulse commences immediately after the static friction dissipates and therefore the remaining pulse height after an initial start-up pulse can be reduced much less than that required to initiate motion.

This type of pulse was used for the experiments in this paper.

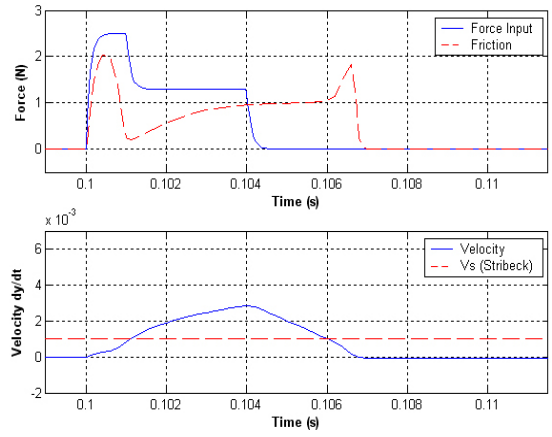


Figure 5: Modified pulse with 2ms start up pulse to overcome static friction and initiate motion.

2.3 Performance at Very Low to High Velocity Regimes

This section investigates how the hybrid PID + impulse controller performs at higher velocities exceeding the Stribeck threshold of approximately .09 rads. For this region of velocities, the highly non linear static and negative viscous friction components are substantially reduced relative to the total and the Coulomb and viscous frictions become the dominant resisting friction. For these velocities, the conventional linear PID controller is well suited. Subsequently, the addition of an impulse torque request may be deleterious to the servomechanism's performance in the region of higher velocity.

Figure 6 shows a series of varying ramp responses from 0.02 rad/s up to 0.35 rad/s using the friction test bed. The range of speeds ensures that the mechanism is operating in both the nonlinear and linear friction regions. Figure 7 compares the Mean Value of the Absolute Error (MAE) for each speed from 7 to 10 seconds respectively. A standard form for MAE is (Ogata, 1990):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left(\left|x - x_i\right|\right) \qquad (4)$$

Where n is the number of data points, x_i is the mechanism position, and x is the reference position.

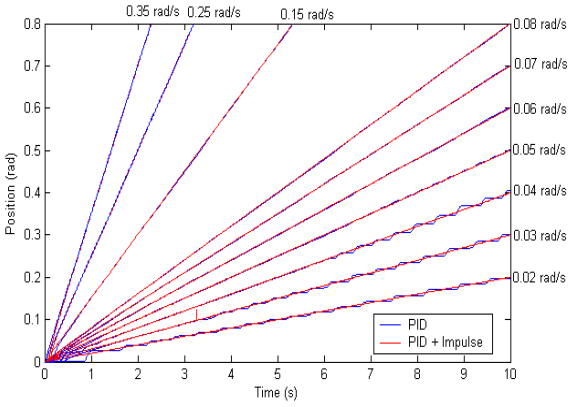


Figure 6: Tracking response for the friction test bed using PID and PID + impulse controllers for varying position ramps (0.02 rad/s to 0.35 rad/s).

For the velocities below the Stribeck velocity threshold, the hybrid PID + impulse controller significantly outperforms the conventional PID controller. However, as the velocity increases, the mean errors of both the PID and PID + impulse controllers begin to converge ($\omega \approx 0.15$rad/s), and at a

critical velocity above the Stribeck region, the PID controller becomes more precise. This increase in precision for the PID controller can be expected since for this higher range of velocities, a conventional linear PID controller will sufficiently counteract the linear fiction without the need of any additional torque. These experiments show that combining the impulse action for high range velocities can be unnecessary and in some instances counterproductive. One way to avoid this loss of performance using a hybrid controller is to disable the impulse torque request at higher velocities to allow the PID part to work autonomously.

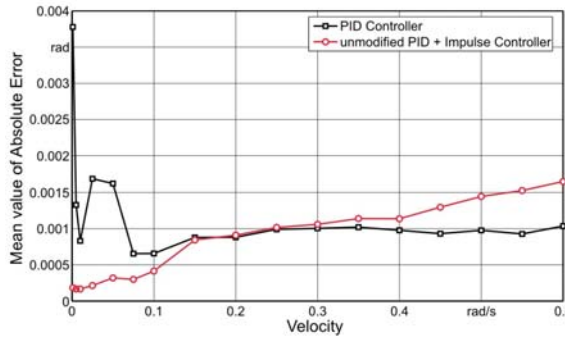


Figure 7: Mean value of the absolute error for each of the position tracking ramps shown in Figure 6 for the period 7 – 10 seconds.

3 TRANSITION VELOCITY CONTROLLER

This section evaluates a series of transition velocity controller functions which disable or limit the impulsive controller above the critical Stribeck velocity.

3.1 On/Off Control using the Critical Velocity

Here the controller's impulsive part is switched off at the critical velocity. The set of conditions for which this occurs is simply defined by the following:

$$\begin{aligned} &\text{If } y(v) \leq \text{ critical velocity, then the impulse} \\ &\text{force} \quad f_p = \text{constant} \\ &\text{Otherwise } f_p = zero \end{aligned} \tag{5}$$

Where *y(v)* is determined by differentiating the mechanism's actual position.

The initial assumption was that this action would make the system unstable at this moment. Close inspection of a velocity tracking task (Figure 8) confirms the mechanism cyclically overshooting and undershooting. This results in the error of the position tracking task increasing for the critical velocity and velocities nearby. This can be seen in Figure 9 between the velocity range of 0.25 and 0.35 rad/s.

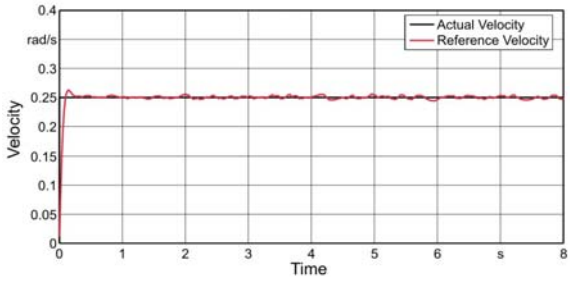


Figure 8: Velocity response when tracking defined disabling velocity of v=0.25 rad/s.

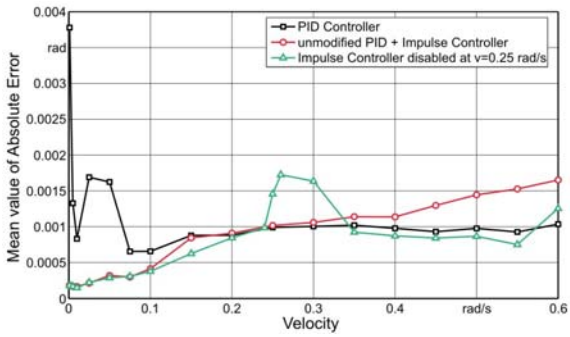


Figure 9: Mean value of the absolute error of the unmodified controllers and the modified PID plus Impulse controller with a disabled impulsive part with respect to the actual velocity.

The loss of torque from the impulsive controller immediately affects the mechanism and the PID controller cannot counteract this quickly enough. The loss of torque causes the velocity to drop under the critical velocity and the impulsive part is immediately enabled again. This makes the system unstable and the controller cyclically enables and disables the impulse controller. However, at velocities above this transition region the position tracking error is consistent with the PID only controller, as expected.

It is clear that using only servomechanism velocity output as a function to control the transition, cannot be an option if tracking accuracies near critical velocities are to be maintained. A solution to this is to transition the controller as a function of

reference velocity rather than system velocity as shown in Figure 10.

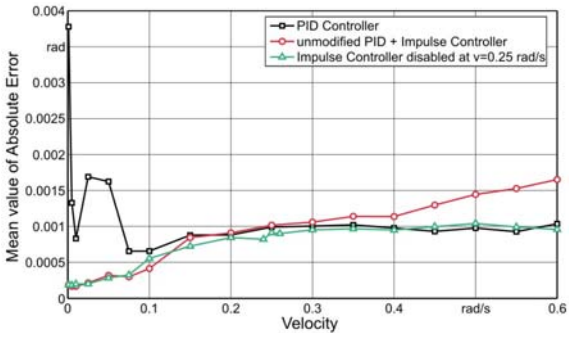


Figure 10: Mean value of the absolute error of the unmodified controllers and the modified PID plus Impulse controller with a disabled impulsive part with respect to the reference velocity.

3.2 Sinusoidal Reference Position Tracking

To further trial the modified controller, an additional experiment tested the system's ability to track changing velocities that pass through the critical velocity regime. In this case, a sinusoidal position reference ensures a continuous change in velocities for both positive and negative accelerations. By using the Integral Absolute Error (IAE) criterion, the error of a statistically relevant series of position trace experiments can be calculated, and a performance measure between each controller established. A standard measure is given by (Ogata, 1990):

$$\int_0^\infty |e(t)|\,dt \tag{6}$$

Where *e(t)* is the error with respect to time *t*.

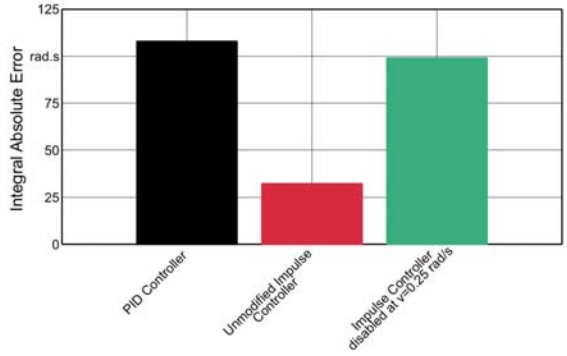


Figure 11: The IAE of the unmodified controllers and the on/off modified controller.

Figure 11 shows a comparison of the results where there is clearly no improvement in accuracy for the modified PID + impulse controller over the original hybrid controller for these conditions of changing velocity. A subsequent breakdown of the error with respect to time shows in Figure 12 that the modified controller mostly counter-performs during acceleration but also partly during deceleration.

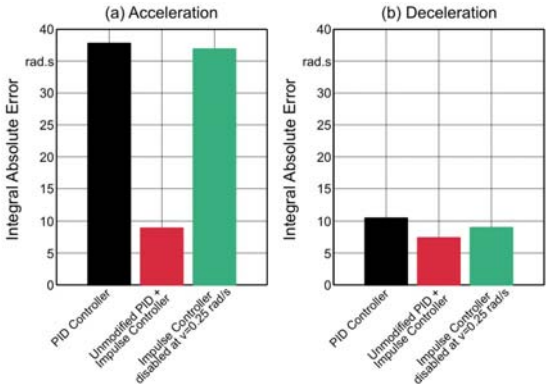


Figure 12: The IAE of the unmodified controllers and the on/off modified controller during (a) acceleration; and (b) deceleration while tracking the sinusoidal position curve.

A closer examination of the position trace (Figure 13) shows that the loss of torque during the point of disabling creates a torque deficiency which the conventional PID controller struggles to correct in a reasonable time frame. A proposed solution to this is to replace the instantaneous on/off switching function with a linear decaying ramp so that abrupt impulse torque removal is avoided and instead gradually transitioned.

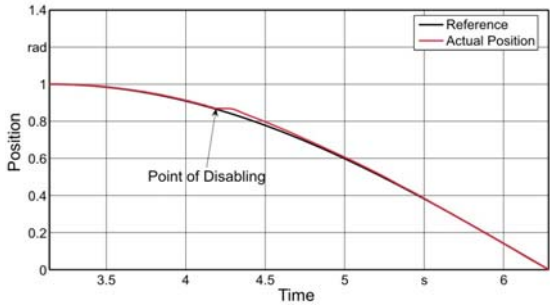


Figure 13: Comparison of the reference and actual position during acceleration while tracking a sinusoidal position input.

3.3 Transition Velocity Function

A solution to transitioning the impulse torque output from unity gain to zero, is given by the following simple linear function:

$$y(v) = a \cdot v + b \quad (7)$$

Where v is the reference velocity of the system and the constants a and b are experimentally determined by trial over a range of velocities either side of the system's critical velocity.

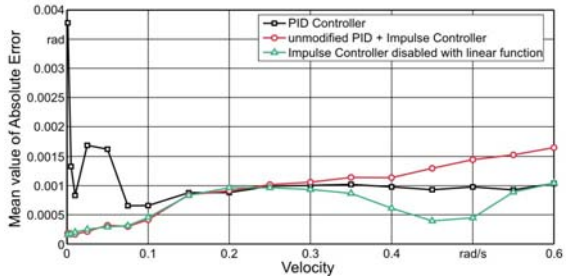


Figure 14: MAE for constant velocity position tracking tasks for the unmodified and linear function transitional impulse controllers.

Figure 14 compares the MAE for each controller. Surprisingly, the linear transition function even improves the accuracy in position at the intermediate higher velocities before the impulse torque is fully transitioned to zero. This improved performance shows that the controller accuracy can be noticeably improved by limiting the pulse height at higher velocities instead of disabling it.

Subsequently, a new function with modified requirements was determined. The function provides:

- A fast reduction in pulse height matched to the PID sampling rate when switching from full to partial impulse control;
- Rather than disabling the full impulse completely it instead reduces it to a fraction of the original pulse height;
- Control over the magnitude of impulse for either acceleration or deceleration regimes.

All of these requirements can be realised with an exponential function. The basic equation used was:

$$y(v) = e^{a(v-b)} + c \quad (8)$$

The parameter a was chosen to be -10 as the exponential function should be designed to have a negative slope that simultaneously reduces the impulse height rapidly. The parameters b and c are determined through boundary conditions as follows:

$$y(v = 0.15) = e^{-10(v-b)} + c = 1 \quad (9)$$

$$y(v = 0.35) = e^{-10(v-b)} + c = 0.25 \quad (10)$$

Solving these equations gives:

$$y(v) = e^{-10(v-0.1358)} + 0.1326 \quad (11)$$

Where v is the reference velocity given by the tracking task. The boundary conditions are selected by trial using a range of varying pulse heights.

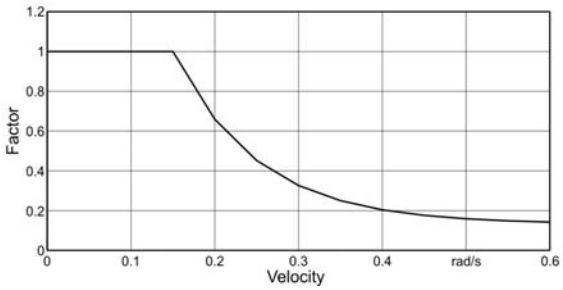


Figure 15: Exponential function for transitioning the limiting of the impulse torque with respect to velocity.

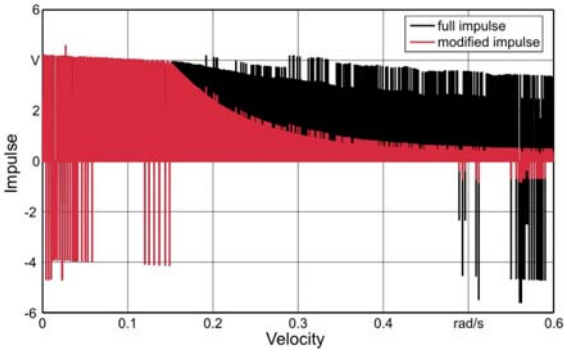


Figure 16: Typical controller torque command for a full impulse and modified impulse applied to friction test bed.

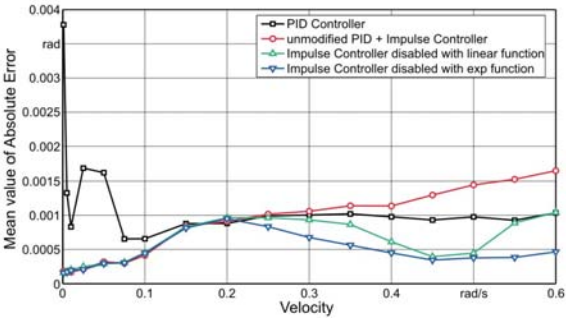


Figure 17: MAE for constant velocity position tracking tasks for the unmodified and transitional impulse controllers.

Figure 15 shows a graphical representation of the exponential function, while Figure 16 gives an example of a typical controller torque command.

The exponential function was shown to provide a significant improvement in the accuracy of velocity tracking (Figure 17). However, further velocity tracking experiments showed that the improvement can only be achieved when tracking constant velocities and is particularly counterproductive during acceleration and varying deceleration. This is caused by the insufficient response of the controller to change the pulse height relative to the rapid changes in velocity.

A solution for ensuring a smooth transition between different tracking tasks with different pulse heights is a time dependant exponential function with additional conditions. If F_1 is the factor determined by the modification done in the previous section, the requirements for the new function are as follows:

$$f(t=0)=\frac{1}{F_1} \tag{12}$$

$$f(t=\infty)=1 \tag{13}$$

This leads to the following equation:

$$f(t)=e^{-0.5\times t+\ln\left(\frac{1}{F_1}-1\right)}+1 \tag{14}$$

Multiplying this equation with Equation 11 gives a smooth transition between the full impulse height during the acceleration and the fraction of the pulse height during constant velocity after acceleration.
To compare each controller for a range of conditions and to test the controller's stability, a varying position tracking experiment was devised with the resulting trace shown in Figure 18. This trajectory was chosen as a demanding trajectory including several velocity reversals and various velocity gradients.

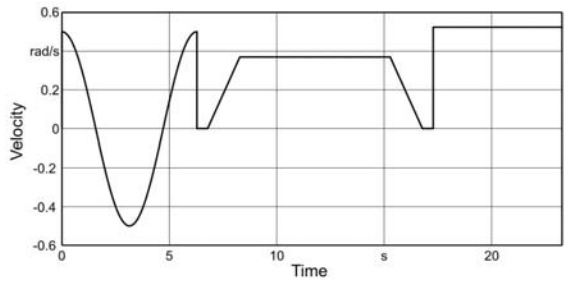


Figure 18: Position tracking task for testing stability.

After repeating the experiments for each controller, the IAE criterion was used to compare each controller and the results shown in Figure 19. The results clearly show a marked improvement in the overall accuracy of the system when using the impulse controller with a time varying exponential function to transition the impulse torque during acceleration and deceleration. Furthermore, the results show that the controllers are robust enough to remain stable over the fairly demanding range of reference conditions tested.

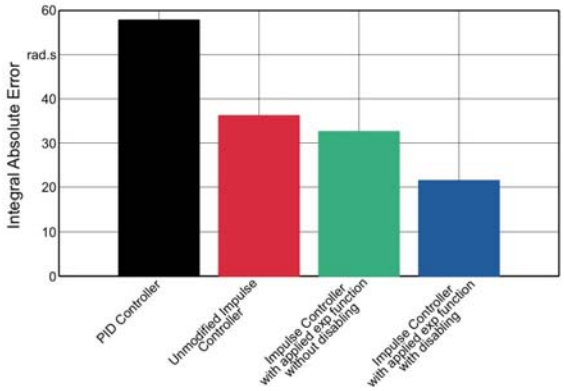


Figure 19: The IAE for the velocity tracking using the unmodified controller, exponential function and exponential function with time varying transition.

3.4 Discussion of Results

This set of results demonstrates the impulse transitional velocity function can be successfully applied to a servomechanism, having characteristics of high non-linear friction. The results show that the unmodified impulse controller significantly outperforms the conventional PID controller at very low velocities. However, as the velocity increases, the mean errors of both the PID and PID + impulse controllers begin to converge above the Stribeck region and the PID controller becomes more precise.

By applying an exponential function which includes consideration of time dependent boundary conditions, the impulse controller can be transitionally reduced to exploit the robustness of a conventional PID controller at higher velocities where viscous friction dominates.

A comparison of the Mean Value of the Absolute Error and the Integral of the Absolute Error for each controller shows that the impulse controller with the velocity dependant exponential function for impulse torque transitioning achieved a more precise result. This controller was proven to be robust enough to maintain stability during a rigorous position tracking task.

4 CONCLUSIONS

Advances in digital control have allowed the power electronics of servo amplifiers to be manipulated in a way that will improve servomechanism precision without modification to the mechanical plant.

A previously developed hybrid PID + Impulse controller which does not require the mechanism to come to a complete stop between pulses has been modified to further improve accuracy in the presence of stick-slip friction. This modification transitions the decay of the impulse torque command at higher velocities. Many experimental tests showed that this innovation provided substantial additional improvement in the mechanism's position accuracy in comparison with other control strategies. This has been demonstrated on a servomechanism which is typical of those used to control each axis of industrial mechanisms such as a robot arm.

Future work is proceeding on optimising the parameters using a method generic to any mechanism, which does not rely on trial and error and is applicable to a greater rang of trajectories.

REFERENCES

Armstrong-Hélouvry, B., 1991, "*Control of Machines with Friction*" Kluwer Academic Publishers, 1991, Norwell MA.

Armstrong-Hélouvry, B., Dupont, P., and Canudas de Wit, C., 1994, "*A survey of models, analysis tools and compensation methods for the control of machines with friction*" *Automatica*, vol. 30(7), pp. 1083-1138.

Canudas de Wit, C., Olsson, H., Åström, K. J., 1995 "*A new model for control of systems with friction*" I*EEE Tansactions on Automatic Control*, vol. 40 (3), pp. 419-425.

Dahl, P., 1968, "*A solid friction model*" Aerospace Corp., El Segundo, CA, Tech. Rep. TOR-0158(3107-18)-1.

Dahl, P, 1977, "*Measurement of solid friction parameters of ball bearings*" Proc. of 6th annual Symp. on Incremental Motion, *Control Systems and Devices*, University of Illinois, ILO.

Hojjat, Y., and Higuchi, T., 1991 "*Application of electromagnetic impulsive force to precise positioning*" *Int J. Japan Soc. Precision Engineering*, vol. 25 (1), pp. 39-44.

Johannes, V. I.., Green, M.A., and Brockley,C.A., 1973, "*The role of the rate of application of the tangential force in determining the static friction coefficient*", *Wear*, vol. 24, pp. 381-385.

Johnson, K.L., 1987, "*Contact Mechanics*" Cambridge University Press, Cambridge.

Kato, S., Yamaguchi, K. and Matsubayashi, T., 1972, "*Some considerations of characteristics of static friction of machine tool slideway*" *J. o Lubrication Technology*, vol. 94 (3), pp. 234-247.

Li, Z, and Cook, C.D., 1998, "*A PID controller for Machines with Friction*" Proc. Pacific Conference on Manufacturing, Brisbane, Australia, 18-20 August, 1998, pp. 401-406.

Ogata, K., *Modern Control Engineering*. 1990, Englewood Cliffs, New Jersey: Prentice Hall.

Olsson, H., 1996, "*Control Systems with Friction*" Department of Automatic Control, Lund University, pp.46-48.

Popovic, M.R., Gorinevsky, D.M., Goldenberg, A.A., 2000, "*High precision positioning of a mechanism with non linear friction using a fuzzy logic pulse controller*" *IEEE Transactions on Control Systems Technology*, vol. 8 (1) pp. 151-158.

Rabinowicz, E., 1958, "*The intrinsic variables affecting the stick-slip process*," *Proc. Physical Society of London*, vol. 71 (4), pp.668-675.

Richardson, R. S. H., and Nolle, H., 1976, "*Surface friction under time dependant loads*" *Wear*, vol. 37 (1), pp.87-101.

van Duin, S., 2006, *"Impulse Control Systems for Servomechanisms with Nonlinear Friction"*, Ph.D. dissertation, University of Wollongong.

van Duin, S, Cook, C., Li, Z., Alici, G., 2007, "*A modified impulse controller for Improved Accuracy of Robots with Friction*", Proceedings International Conference on Informatics in Control, Automation and Robotics, Angers, France.

Yang, S., Tomizuka, M., 1988, "*Adaptive pulse width control for precise positioning under the influence of stiction and Coulomb friction*" *ASME J .od Dynamic Systems, Measurement and Control*, vol. 110 (3), pp. 221-227.

ON THE WORK-IN-PROCESS CONTROL OF PRODUCTION NETWORKS

Nikos C. Tsourveloudis
Machine Tools Laboratory, Technical University of Crete, Chania, Greece
nikost@dpem.tuc.gr

Keywords: Production Networks, Work-In-Process, Fuzzy Control, Evolutionary Algorithms, Controller Design.

Abstract: The effectiveness of evolutionary optimized fuzzy controllers for production scheduling has been proven in the past. The objective of the control/scheduling task in this context, is to continuously adjust the production rate in a way that: 1) satisfies the demand for final products, 2) keeps the inventory as low as possible. The evolutionary optimization identifies fuzzy control solutions which simultaneously satisfy those restrictions. The important question here is: How robust and generic is the outcome of the evolutionary process? In this paper we face this question by testing the evolutionary tuned fuzzy controllers under several demand patterns, as the actual demand might be different from those used for evolution\optimization. Extensive simulations of a supervisory controller identify the performance of the evolutionary-fuzzy strategy in comparison to a pure knowledge based one.

1 INTRODUCTION

As the manufacturing industry moves away from the mass production paradigm towards the agile manufacturing, the life cycle of products gets shorter while the need for a wide variety of them increases. Keeping large inventories in stock tends to be unattractive in today's markets. The same holds for the unfinished parts throughout the manufacturing system, widely known as Work-In-Process (WIP), as it represents an already made expense with unknown profitability due to the rapidly changing demand. In a highly changing demand environment, the accumulated inventories are less desirable than ever.

The work-in-process inventory is measured by the number of unfinished parts in the buffers throughout the manufacturing system and it should stay as small as possible (Conway et al., 1998), (Bai and Gershwin, 1994).

Traditionally, inventory control methods in this field can be roughly grouped into mathematical modelling approaches, computerized planning methods, such as material requirement planning (MRP), and heuristic scheduling strategies. Many control policies (CONWIP-constant WIP, base stock method etc.|) aim in keeping WIP at low levels (Gershwin, 1994). However, an exact optimal value of WIP cannot be determined in realistic manufacturing conditions. Therefore, the problem of WIP determination and control is amenable to an artificial intelligent treatment, as suggested in (Custodio et al., 1994), (Tsourveloudis et al., 2000) and recently in (Ioannidis et al., 2004) and (Tsourveloudis et al., 2006). The supervisory controller suggested in (Ioannidis et al., 2004) is used to tune a set of lower-level distributed fuzzy control modules that reduce WIP and synchronize the production system's operation. The overall control objective is to keep the WIP and cycle time as low as possible, while maintaining quality of service by keeping the backlog to an acceptable level.

Fuzzy logic has been used in tandem to Evolutionary Algorithms (EA) so as to keep the WIP and cycle time as low as possible, and at the same time to maintain high utilization (Tsourveloudis et al. 2006, Tsourveloudis et al., 2007). The objective in those works was to optimize the control policy in a way that satisfies the (random) demand for final products while keeping minimum WIP within the production system. During the evolution, the EA identifies those set of parameters for which the fuzzy controller has an optimal performance with respect to WIP minimization for several demand patterns.

The use of evolving genetic structures for the production scheduling problem, has recently gained a lot of acceptance in the automated and optimal design of fuzzy logic systems (Tedford and Lowe, 2003, Gordon et al. 2001). However, a potential

problem is that the evolutionary (or genetically) evolved fuzzy controllers might perform optimal only under the conditions involved in the evolution process. In this paper we examine the performance of evolutionary optimized controllers in contrast to heuristically designed fuzzy controllers. For comparisons purposes we test the controllers in conditions different from the ones they have been designed for. In this way, some useful insights regarding the design robustness of the evolutionary tuned fuzzy controllers may be drawn.

The rest of the paper is organized as follows. Section 2 describes the evolutionary fuzzy scheduling concept that is used for WIP minimization. Two control approaches are presented: the *distributed* and the *supervised* one. Section 3 describes the comparison scenarios and presents experimental results for production lines and networks. Issues for discussion and remarks as well as suggestions for further development are presented in the last section.

2 EVOLUTIONARY-FUZZY SCHEDULING

A production network consists of machines (operation stations) and buffers (storage areas). Items are received at each machine and wait for the next operation in a buffer with finite capacity. WIP may increase because of unanticipated events, like machine breakdowns and potential consequent propagation of these events. For example, a failed machine with operational neighbours forces to an inventory increase of the previous storage buffer. If the repair time is big enough, then the broken machine will either block the previous station or starve the next one. This "bottleneck" effect will propagate throughout the system.

Clearly, production scheduling of realistic manufacturing plants must satisfy multiple conflicting criteria and also cope with the dynamic nature of such environments. Fuzzy logic offers the mathematical framework that allows for simple knowledge representations of the production control/scheduling principles in terms of IF-THEN rules. The expert knowledge that describes the control objective (that is WIP reduction) can be summarized in the following statements (Tsourveloudis et al, 2000, Tsourveloudis et al., 2006):

If the surplus level is satisfactory then try to prevent starving or blocking by increasing or decreasing the production rate accordingly,

else

If the surplus is not satisfactory that is either too low or too high then produce at maximum or zero rate respectively.

In fuzzy logic controllers (FLCs), the control policy is described by linguistic IF-THEN rules similar to the above statements. The essential part of every fuzzy controller is the knowledge acquisition and the representation of the extracted knowledge with certain fuzzy sets/membership functions. Membership functions (MFs) represent the uncertainty modelled with fuzzy sets by establishing a connection between linguistic terms (such as low, negative, high etc) and precise numerical values of variables in the physical system. The correct choice of the MFs is by no means trivial and plays a crucial role in the success of an application. If the selection of the membership functions is not based on a systematic optimization procedure then the adopted fuzzy control strategy cannot guarantee minimum WIP level.

The evolutionary-fuzzy synergy attempts to minimize the empirical/expert design and create MFs that fit best to scheduling objectives (Tsourveloudis et al., 2006). In this context, the design of the fuzzy controllers (distributed or supervisory) can be regarded as an optimization problem in which the set of possible MFs constitutes the search space. Evolutionary Algorithms (EAs) are seeking optimal or near optimal solutions in large and complex search spaces and therefore have been successfully applied to a variety of scheduling problems with broad applicability to manufacturing systems (Tedford and Lowe, 2003). The objective is to optimize a performance measure which in the EAs context is called fitness function. In each generation, the fitness of every chromosome is first evaluated based on the performance of the production network system, which is controlled through the membership functions represented in the chromosome. A specified percentage of the better fitted chromosomes are retained for the next generation. Then parents are selected repeatedly from the current generation of chromosomes, and new chromosomes are generated from these parents. One generation ends when the number of chromosomes for the next generation has reached the quota. This process is repeated for a pre-selected number of generations.

2.1 Distributed Evolutionary-fuzzy Control

The architecture of the distributed evolutionary-fuzzy WIP control scheme is extensively discussed

in (Tsourveloudis et al., 2006) and (Tsourveloudis et al., 2007). The control objective of the distributed scheduling approach, as earlier stated, is to satisfy the demand and, at the same time, to keep WIP as low as possible. This is attempted by regulating the processing rate r_i at every time instant. The processing rate r_i of each machine at every time instant is:

$$r_i' = \mathrm{f}_{\mathrm{IS}}(b_{j,i}, b_{i,l}, x_i, s_i) = \begin{cases} 0 & \text{if } s_i = 0 \\ \dfrac{\sum r_i \mu_R^*(r_i)}{\sum \mu_R^*(r_i)} & \text{if } s_i = 1 \end{cases}, \quad (1)$$

where, $\mathrm{f}_{\mathrm{IS}}(b_{j,i}, b_{i,l}, x_i, s_i)$ represents a fuzzy inference system that takes as inputs the level $b_{j,i}$ of the upstream buffer, the downstream buffer level $b_{i,l}$, x_i is the surplus (cumulative production minus demand) and s_i is a non fuzzy variable denoting the state of the machine, which can be either 1 (operative) or 0 (stopped).

The fitness function $F(x_i)$ of each individual x_i, which associates the demand with the cumulative production of the manufacturing system is:

$$F(x_i) = \left[\sum_{j=1}^{N} (D(t_j) - PR(t_j))^2 \right]^{-1} \quad (2)$$

where, t is the current simulation time, $D(t)$ is the overall demand and $PR(t)$ is the cumulative production of the system.

As earlier stated, the objective of the evolution process is to optimize the shape of the fuzzy membership function. Indeed, after the evolution process the shape of the membership functions is altered. The best individual is considered to be the one with the biggest fitness. The fittest individuals are selected and they undergo mutations. The fittest controllers and their mutated offsprings are forming the new population. After some generations the algorithm converges and the best individuals represent near optimal solutions.

2.2 Supervised Evolutionary-fuzzy Control

In control systems literature a supervisor is a controller (supervisory controller) that utilizes available data to characterize the overall system's current behavior, potentially modifying the lower level controllers to ultimately achieve desired specifications. The supervisory controller in this, and also in our past works, is used to tune the distributed controllers in a way that improves performance without dramatic changes in the structure of the control architecture, as justified in (Ioannidis et al., 2004). The concept of the supervised evolutionary-fuzzy WIP control scheme is shown in Figure 1. The fitness function in the supervisory approach case was chosen to be the following:

$$F = \left(c_I \overline{WIP} + c_b \overline{BL} \right)^{-1} \quad (3)$$

where, $\overline{WIP}$ and $\overline{BL}$ are the mean work-in-process and mean backlog (=cumulative production minus demand), respectively. The c_I, c_b are weighting factors that represent the unit costs of inventory and backlog, respectively. Assuming that the capacity of a production system is given, equations (2) and (3) show that the evolved MFs are highly based (in terms of their support and shape) on the demand values. Obviously, the value of demand is crucial for WIP and backlog determination in (3). Some of the questions arise here concerning demand, are:

- What happens when actual demand is different (in both magnitude and changing pattern) than the one considered during controller's evolution?
- Is the evolved controller robust enough to absorb random variations of demand?
- Does the original (without MF optimization) heuristic fuzzy controller perform better in unknown demands?

Since there are no analytical solutions to those questions, in what follows we will examine and compare the performance of both evolutionary and heuristic fuzzy controllers through simulation, for a variety of test cases.

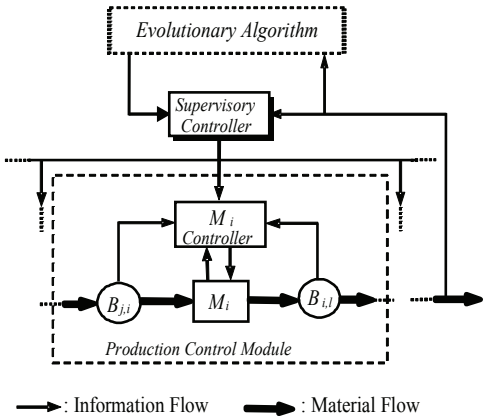


Figure 1: Supervisory control: Evolutionary-fuzzy concept.

3 TESTING AND RESULTS

The evolutionary-fuzzy approaches suggested in (Tsourveloudis et al., 2007), are tested and compared to the heuristic fuzzy approaches initially suggested in (Tsourveloudis et al., 2000). In the all simulations performed we assume that the machines fail randomly, with a failure rate p_i. This rate is known and set before the simulation starts. Also, machines are repaired randomly with rate rr_i. The resources needed for repairs are assumed to be available. The times between failures and repairs are exponentially distributed. All machines operate at known, but not necessarily equal rates. Each machine produces in a rate $r_i \leq \mu_i$, where μ_i is the maximum processing rate of machine M_i. We also assume that the flow of parts within the system is continuous.

In the production network shown in Figure 2, the circles represent buffers and squares are machining stations. This network is identical to the one discussed in previous works (Tsourveloudis et al., 2000, 2006a, 2006b, 2007). For simplicity it is assumed that this network produces one part type. Lines and networks producing multiple part types have been discussed in (Tsourveloudis et al., 2000), (Ioannidis et al., 2004) and it has been shown that have similar behavior to the single-part-type systems. One important observation made in (Ioannidis et al. 2004, Tsourveloudis et al. 2006, 2007) was that the evolutionary tuned fuzzy controllers achieved a substantial reduction of WIP in almost all test cases.

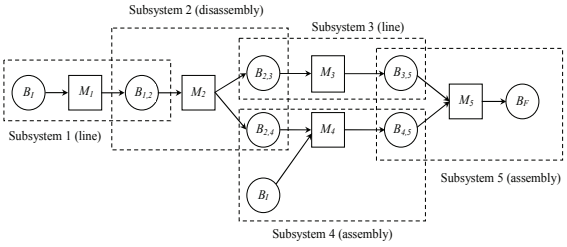


Figure 2: Layout of the production network.

Here we further investigate the performance of the evolutionary tuned fuzzy controllers, keeping unaltered the controllers' design but with demand patterns that are significantly changed. In practice, demand is the main uncertainty of almost all production system/networks. Changes in demand may cause significant problems in balancing production lines

3.1 Supervised Control of Networks

The objective is to examine the robustness of the supervised control approach. The simulation testbed used for this test case was developed in SIMULINK and its main blocks are shown in Figure 3.

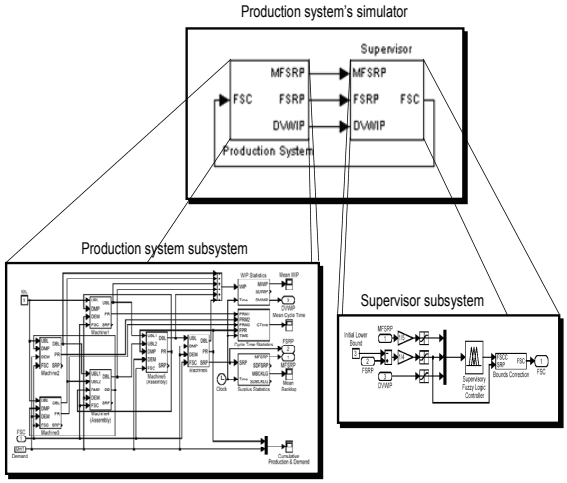


Figure 3: SIMULINK model of the supervisory control.

The performance of the evolutionary-fuzzy supervised approach was examined for various demand patterns other than the one used during the optimization of the membership functions. During the evolution procedure, demand was considered either one (one product per time unit) or zero (no demand at the time unit) and the selection between those two values was triggered in a random order. During our testing different demands were used:

Demand Pattern 1 (DP1): The system accepts orders of 1 product per time unit. The time unit is set equal to 0.05 of the simulation step. This is similar to the demand pattern used for the optimization of the controller.

Demand Pattern 1.5 (DP1.5): The system accepts orders for 1.5 products per time unit, which is set 0.05 of the simulation step.

Demand Pattern 3 (DP3): The system accepts orders for 3 products per time unit, which is also set 0.05 of the simulation step.

Figure 4 presents the mean WIP and Backlog for the above mentioned demand patterns. As it can be seeing, the mean WIP of DP1 is higher than the other two demands, but it fully satisfies the requested demand. DP1.5 and DP3 fail in satisfying demand in the same test run. It also can be seen, in Figure 4, that when the demand is 3 times higher (DP3) than the one used for the evolution (DP1), then it cannot be satisfied as the backlog accumulates rapidly (DP3-BL).

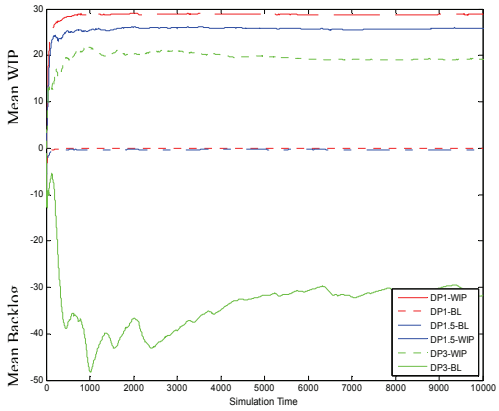


Figure 4: WIP and Backlog levels of the supervisory control for various demand sizes.

However, when demand is increased for 50%, (DP1.5) the unsatisfied demand (DP1.5-BL) is almost zero which shows that the supervisor works satisfactorily for demand changes of this magnitude: +50% of the demand used during the evolution of the fuzzy supervisory controller. This important observation was also noted through a series of simulation runs for demands lower than the one used in the evolution. In this case, a slight increase in the mean WIP levels was observed.

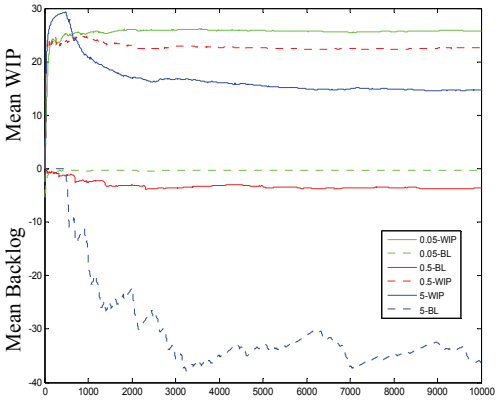


Figure 5: WIP and Backlog levels for changing demand rates.

Not only the magnitude but also the frequency of demand changing was examined. Figure 5 presents the WIP and Backlog mean levels when the DP1.5 demand pattern changes every 0.05, 0.5, and 5 time units respectively. It can be observed that in lower demand rates the controller keeps the backlog orders close to zero, while in higher rates although the controller keeps WIP in low levels, fails in satisfying the demand (5-BL in Figure 5).

4 CONCLUSIONS

WIP itself cannot represent adequately of production system's performance. One has to take into account also the accumulated orders backlog. It is also known that when demand is very high one may consider that service rate and thus backlog is more important than WIP. When demand can be easily satisfied and backlog is in low levels, a substantial reduction of WIP may be more important than a small increase in backlog. What we have seen so far is that with the aid of the evolutionary-fuzzy controllers the system's performance becomes more balanced in terms of mean WIP and backlog. WIP is substantially reduced in the evolutionary-fuzzy approach compared to the empirical selected fuzzy controllers. The same observation holds for the supervisory control of production networks where significantly increased demands were accommodated.

The heuristic fuzzy control approach cannot achieve the performance of the evolutionary-fuzzy. However, it is still better than previously reported "bang-bang" control approaches. Even when compared to the evolutionary-fuzzy approach it is much simpler in the design process as it steps on the human expertise/knowledge regarding the production system. In others words, one should very fast design, built and put to work a fuzzy controller with membership functions that represent the expert knowledge in contrast to the evolutionary-fuzzy system whose parameters are automatically set by the optimization procedure.

The evolutionary-fuzzy controllers are capable of maintaining low WIP levels for product demands other than the ones used during the optimization. Therefore, the evolutionary algorithms clearly represent a successful approach towards the optimization of robust scheduling approaches.

REFERENCES

Conway R., Maxwell W, McClain J.O., Joseph Thomas L, 1998. The role of work-in-process inventory control: single-part-systems. Oper. Res. 36, pp. 229-241.

Bai S.X. and Gershwin S.B., 1994. Scheduling manufacturing systems with work-in-process inventory control: multiple-part-type systems. Int. J. Prod. Res. 32, 365-386.

Gershwin S.B ., 1994. *Manufacturing Systems*

Engineering. Prentice Hall, New Jersey.
Custodio L., Sentieiro J., Bispo C., 1994. Production planning and scheduling using a fuzzy decision system. IEEE Trans. Robot. Automat. 10, pp. 160-168.
Tsourveloudis N.C., Dretoulakis E., Ioannidis S., 2000. Fuzzy work-in-process inventory control of unreliable manufacturing systems. Inf. Sci. 27, pp. 69-83.
Ioannidis S., Tsourveloudis N.C., Valavanis K.P., 2004. Fuzzy Supervisory Control of Manufacturing Systems. IEEE Trans. Robot. Automat. 20, pp. 379-389.
Tsourveloudis N. C., L. Doitsidis, S. Ioannidis, 2006. Work-in-Process Scheduling by Evolutionary Tuned Distributed Fuzzy Controllers. In: Proceedings of the IEEE International Conference on Robotics and Automation, Orlando, FL, USA.
Tsourveloudis N. C., S. Ioannidis, K. P. Valavanis, 2006. Fuzzy Surplus based Distributed Control of Manufacturing Systems. Advances in Production Engineering and Management. 1, pp. 5-12.
Tsourveloudis N. C, Doitsidis L., Ioannidis S., 2007. Work-In-Process Scheduling by Evolutionary Tuned Fuzzy Controllers. International Journal of Advanced Manufacturing Technology. 34, no. 7-8, pp. 85-97.
Tedford J.D. and Lowe C., 2003. Production scheduling using adaptable fuzzy logic with genetic algorithms. Int. J. Prod. Res. 41, pp. 2681–2697.
Gordon O., Herrera F., Hoffmann F., Luis M., 2001. *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Base.* World Scientific Publishing Co. Pte. Ltd, U.K.

BATCH REINFORCEMENT LEARNING
An Application to a Controllable Semi-active Suspension System

Simone Tognetti, Marcello Restelli, Sergio M. Savaresi
Dipartimento di elettronica e informazione, Politecnico di Milano, via Ponzio 34/5, 20133 Milano, Italy
tognetti@elet.polimi.it, restelli@elet.polimi.it, savaresi@elet.polimi.it

Cristiano Spelta
Dipartimento di Ingegneria dellInformazione e Metodi Matematici, Universit degli Studi di Bergamo
viale Marconi 5, 24044 Dalmine (BG), Italy
cristiano.spelta@unibg.it

Keywords: Batch-reinforcement learning, Control theory, Non linear optimal control, Semi-active suspension.

Abstract: The design problem of optimal comfort-oriented semi-active suspension has been addressed with different standard techniques which failed to come out with an optimal strategy because the system is hard non-linear and the solution is too complex to be found analytically. In this work, we aimed at solving such complex problem by applying Batch Reinforcement Learning (BRL), that is an artificial intelligence technique that approximates the solution of optimal control problems without knowing the system dynamics. Recently, a quasi optimal strategy for semi-active suspension has been designed and proposed: the Mixed SH-ADD algorithm, which the strategy designed in this paper is compared to. We show that an accurately tuned BRL provides a policy able to guarantee the overall best performance.

1 INTRODUCTION

Among the many different types of controlled suspension systems (see e.g., (Sammier et al., 2003; Savaresi et al., 2005; Silani et al., 2002)), semi-active suspensions have received a lot of attention since they provide the best compromise between cost (energy-consumption and actuators/sensors hardware) and performance. The research activity on controllable suspensions develops along two mainstreams: the development of reliable, high-performance, and cost-effective semi-active controllable shock-absorbers (Electro-Hydraulic or Magneto-Rheological see e.g., (Ahmadian et al., 2001; Guardabassi and Savaresi, 2001; Valasek et al., 1998; Williams, 1997)), and the development of control strategies and algorithms which can fully exploit the potential advantages of controllable shock-absorbers. This work focuses on the control-design issue for road vehicles.

The design problem of optimal comfort oriented semi-active suspension has been addressed with different standard techniques which failed to came out with an optimal strategy because the system is hard non-linear and the solution is too complex to be found analytically. The literature offers many contributions that provide approximate solutions to the non-linear problem, or alternatively, the non-linearity is partially removed to exploit linear techniques (see e.g., (Karnopp and Crosby, 1974; Sammier et al., 2003)-(Savaresi and Spelta, 2008; Valasek et al., 1998)).

In this work, we aimed at solving the optimal control problem of comfort-oriented semi-active suspension by using Batch Reinforcement Learning (BRL). Developed in the artificial intelligent research field, BRL provides numerical algorithms able to approximate the solution of an optimal-control problem without knowing the system dynamics (see (Kaelbling et al., 1996) and (Sutton and Barto, 1998)). The algorithm is independent from the model complexity and can be trained on the real system without knowing its dynamics. We compared the strategy obtained by BRL with the ones given by the state-of-the-art semi-active control algorithms. We showed that an accurately tuned BRL provides a policy able to guarantee the overall best performance.

The outline of the paper is as follows. In Section 2 the control problem is stated. Section 3 recalls the BRL technique. Section 4 sums up the design of BRL-based control rule. Section 5 motivates the choice of algorithm parameters, section 6 presents

experimental results, and finally, section 7 ends the paper with some concluding remarks.

2 PROBLEM STATEMENT AND PREVIOUS WORK

The dynamic model of a quarter-car system equipped with semi-active suspensions can be described by the following set of differential equations (Williams, 1997):

$$\begin{cases} M\ddot{z}(t) &= -c(t)(\dot{z}(t)-\dot{z}_t(t))-k(z(t)-z_t(t)-\Delta_s)-Mg \\ m\ddot{z}_t(t) &= c(t)(\dot{z}(t)-\dot{z}_t(t))+k(z(t)-z_t(t)-\Delta_s)+ \\ & \quad -k_t(z_t(t)-z_r(t)-\Delta_t)-mg \quad [z_t(t)-z_r(t)<\Delta_t] \\ \dot{c}(t) &= -\beta c(t)+\beta c_{in}(t) \qquad [\hat{c}_{min}\leq c_{in}(t)\leq \hat{c}_{max}] \end{cases} \tag{1}$$

where the symbols in (1) are as follows (see also Figure 1): $z(t), z_t(t), z_r(t)$ are the vertical positions of the body, the unsprung mass, and the road profile, respectively. M is the quarter-car body mass; m is the

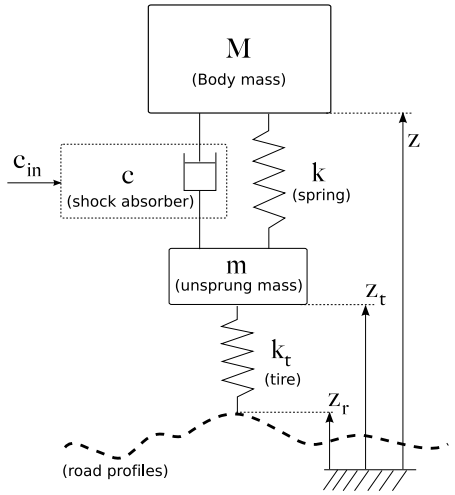


Figure 1: Quarter-car diagram.

unsprung mass (tire, wheel, brake caliper, suspension links, etc.); k and k_t are the stiffnesses of the suspension spring and of the tire, respectively; Δ_s and Δ_t are the lengths of the unloaded suspension spring and tire, respectively; $c(t)$ and $c_{in}(t)$ are the actual and the requested damping coefficients of the shock-absorber, respectively.

The damping-coefficient variation is ruled by a 1st-order dynamic model, where β is the bandwidth; consequently the actual damping coefficient remains in the interval $c_{min} \leq c_{in}(t) \leq c_{max}$, where c_{min} and c_{max} are design parameters of the semi-active shock-absorber. This limitation is the so-called "passivity-constraint" of a semi-active suspension.

For the above quarter-car model, the following set of parameters are used (unless otherwise stated): $M = 400\,kg$, $m = 50\,kg$, $k = 20\,KN/m$, $k_t = 250\,KN/m$, $c_{min} = 300\,Ns/m$, $c_{max} = 3000\,Ns/m$ and $\beta = 100\pi$.

Notice that (1) is non-linear since the damping coefficient $c(t)$ is a state variable; in the case of a passive suspension with a constant damping coefficient c, (1) is reduced to a 4th-order linear system by simply setting $\beta \to \infty$ and $c_{in}(t) = c$.

The general high-level structure of a comfort-oriented control architecture for a semi-active suspension device is the following. The control variable is the requested damping coefficient $c_{in}(t)$. The measured output signals are two: the vertical acceleration $\ddot{z}(t)$ and the suspension displacement $z(t) - z_t(t)$. The disturbance is the road profile $z_r(t)$ (non-measurable and unpredictable signal).

The goal of a comfort-oriented semi-active control system is to manage the damping of the shock-absorber to filter the road disturbance towards the body dynamics. Thus the following cost function is introduced: $J = \int_0^t (\ddot{z}(t))^2 dt$. It has been shown in (Savaresi et al., 2005) that the optimal control strategy is necessarily a rationale that switches from the minimum to the maximum damping of the shock absorber (two-state algorithms).

In the literature there exist many control strategies with a flavor of optimality: Skyhook (SH) (Karnopp and Crosby, 1974) and Acceleration Driven Damping (ADD) (Savaresi et al., 2005). Recently an almost optimal control strategy has been developed: the so-called Mixed SH-ADD (Savaresi and Spelta, 2007). Similarly to SH, also this strategy requires a two-state damper:

$$\begin{cases} c_{in}(t) = c_{max} & \text{if } \left[\ddot{z}^2 - \alpha^2\dot{z}^2 \leqslant 0 \wedge \dot{z}(\dot{z}-\dot{z}_t) > 0\right] \vee \\ & \vee \left[\ddot{z}^2 - \alpha^2\dot{z}^2 > 0 \wedge \ddot{z}(\dot{z}-\dot{z}_t) > 0\right] \\ c_{in}(t) = c_{max} & \text{if } \left[\ddot{z}^2 - \alpha^2\dot{z}^2 \leqslant 0 \wedge \dot{z}(\dot{z}-\dot{z}_t) \leqslant 0\right] \vee \\ & \vee \left[\ddot{z}^2 - \alpha^2\dot{z}^2 > 0 \wedge \ddot{z}(\dot{z}-\dot{z}_t) \leqslant 0\right] \end{cases} \tag{2}$$

Notice that accordingly to the sign of $\ddot{z}^2 - \alpha^2\dot{z}^2$ an appropriate sub-strategy is selected. This quantity is a frequency selector and α represents the desired cross-over frequency between two suboptimal strategies, namely SH and ADD (see. (Savaresi and Spelta, 2007)). A single sensor implementation of this strategy has been recently developed (the so-called 1-Sensor-Mix, (Savaresi and Spelta, 2008)). SH, ADD and Mixed SH-ADD have been already compared in the time and frequency domains (Savaresi and Spelta, 2007).

3 REINFORCEMENT LEARNING

Research in Reinforcement Learning (RL) aims at designing algorithms by which autonomous agents

(controller) can learn to behave (estimation of control policy) in some appropriate fashion in some environment (controlled system), from their interaction (control variable) with this environment or from observations gathered from the environment (see e.g. (Sutton and Barto, 1998) for a broad overview).

The interaction between the agent and the environment is modeled as a discrete-time Markov Decision Process (MDP). An MDP is a tuple $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma >$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \rightarrow \Pi(\mathcal{S})$ is the transition model that assigns to each state-action pair a probability distribution over $\mathcal{S}$, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \Pi(\mathbb{R})$ is the reward function, or cost function, that assigns to each state-action pair a probability distribution over $\mathbb{R}$, $\gamma \in [0,1)$ is the discount factor. At each time step, the agent chooses an action according to its current *policy* $\pi : \mathcal{S} \rightarrow \Pi(\mathcal{A})$, which maps each state to a probability distribution over actions. The goal of an RL agent is to maximize the expected sum of discounted rewards, that is to learn an optimal policy π^* that leads to the maximization of the action-value function, or cost-to-go from each state.

The optimal action-value function $Q^*(s(t), a(t))$ $s(t) \in \mathcal{S}, a(t) \in \mathcal{A}$ is defined by the Bellman equation:

$$Q^*(s(t), a(t)) = \sum_{s(t+\Delta T) \in \mathcal{S}} \mathcal{P}(s(t+\Delta T)|s(t), a(t)) \Big[R(s(t), a(t)) + \gamma \max_{a(t+\Delta T) \in \mathcal{A}} Q^*(s(t+\Delta T), a(t+\Delta T)) \Big] \quad (3)$$

where $R(s,a) = E[\mathcal{R}(s,a)]$ is the expected reward. From the Control Theory perspective this equation represents the optimal cost-to-go, indeed it represents the discrete-time version of the Hamilton-Jacobi-Bellman equation.

In order to manage the huge amount of samples needed to solve real-world tasks, batch approaches have been proposed (Riedmiller, 2005; Antos et al., 2008; Ernst et al., 2005). The main idea is to distinguish between the exploration strategy that collects samples (sampling phase), and the off-line learning algorithm that, on the basis of the samples, computes the approximation of the action-value function (learning phase) that is the solution of the control problem.

3.1 Batch Reinforcement Learning

Let us consider a system having a discrete-time dynamics. If the transition model or the reward function are unknown, we cannot use dynamic programming to solve the control problem. However, we suppose to perform a sampling phase by which a set o samples

$$\mathcal{F} = \{< s(t)^i, a(t)^i, s(t+1)^i, r(t)^i >, i = 1..K\} \quad (4)$$

is obtained from one or more system trajectories generated starting from an initial state, following a given policy.

In the learning phase we used Fitted Q-iteration (FQI, see (Ernst et al., 2005)) that reformulates value function estimation as a sequence of regression problems by iteratively extending the optimization horizon ($Q_N - function$). First $Q_0(s,a)$ is set to 0 then the algorithm iterates over the full sample set $\mathcal{F}$. Given the i-th sample $< s(t)^i, a(t)^i, s(t+1)^i, r(t)^i >$ and the approximation of Q-function at time N (Q_N), the estimation of Q_{N+1} is performed by using Q-learning update rule (Watkins, 1989):

$$Q_{N+1}(s(t)^i, a(t)^i) = (1-\alpha) Q_N(s(t)^i, a(t)^i) + \alpha(r(t)^i + \gamma \max_{a' \in \mathcal{A}} Q_N(s(t+1)^i, a')). \quad (5)$$

For each sample a new one is generated replacing single step rewards with estimated Q-values. This defines a regression problem from $s(t)^i, a(t)^i$ to $Q_1(s(t)^i, a(t)^i)$ that enables the estimation of Q_1. Thereafter, at each iteration N, a new estimation is performed exploiting the approximation at the previous iteration.

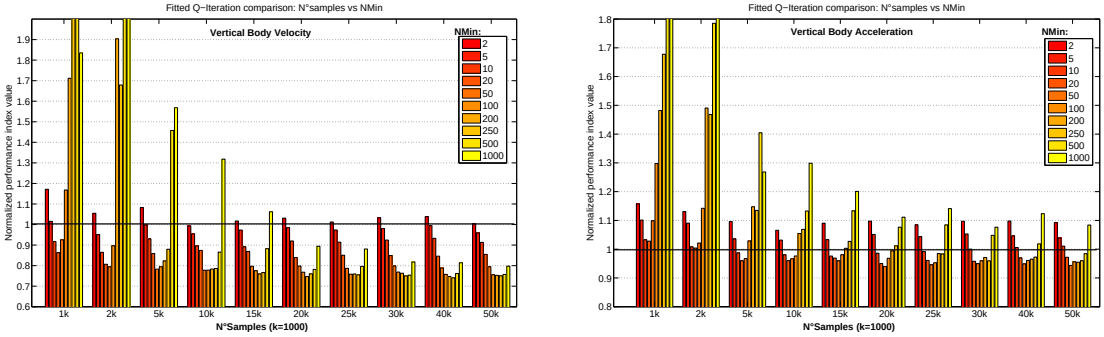
3.2 Q-function Approximation

Tree-based regression methods produce one or more trees (*ensemble*) that are composed by a set of decision nodes used to partition the input space. The tree determines a constant prediction in each region of the partition by averaging the output values of the elements of the training set $\mathcal{TS} = \{(i^1, o^1), \ldots, (i^{\#\mathcal{TS}}, o^{\#\mathcal{TS}})\}$ which belong to this region. Q-function are approximated by considering $i^l =< s(t)^l, a(t)^l >$ while the output o^l is the Q-value of i^l.

We used extremely-randomized tree ensemble (Geurts et al., 2006), that is a regressor composed by a forest of M trees each constructed by randomly choosing K cut-points i_j, representing the j-th component of the action-state space, and the correspondingly binary split $[i_j < t]$, representing the cut-direction. The construction proceeds by choosing a set of tests that maximizes a given score. The algorithm stops splitting a node when the number of elements in the node is lower than a parameter n_{min}.

4 PROBLEM DEFINITION

The state space of a dynamical system is defined by the set of state variables that compound the ODE's system. System 1 in its canonical form (Savaresi



(a) (b)

Figure 2: Comparison of vertical body velocity (J_2) and acceleration (J_1) indexes w.r.t number of samples and NMin. J_1 and J_2 are normalized by Mixed SH-ADD policy ($J_1 = J_2 = 1$).

et al., 2005) has 5 continuous state variables: $\mathcal{S} \equiv < \dot{z}(t), \dot{z}_t(t), \dot{c}(t), z(t) - \bar{z}, z_t(t) - \bar{z}_t >$. The presence of non-measurable road disturbances makes both the transition model and the cost function stochastic.

Since not all the variables are measurable, BRL policy is built exploiting the same sensor measures as those used in the Mixed SH-ADD one (Savaresi and Spelta, 2007): $< \ddot{z}(t), \dot{z}(t), \dot{z}(t) - \dot{z}_t(t) >$. The action space contains only two values: $\mathcal{A} \equiv < c_{in}(t) > | c_{in}(t) \in \{c_{min}, c_{max}\}$.

The minimization of the squared vertical accelerations can be defined as the maximization of the following reward function:

$$\mathcal{R}_1(s(t), a(t)) = -\ddot{z}^2(t + \Delta T). \quad (6)$$

The ideal goal of a semi-active suspension system is to negate the body vertical movements around its steady state conditions, with respect to any road disturbance. Thus, we considered also the following reward:

$$\mathcal{R}_2(s(t), a(t)) = -\dot{z}^2(t + \Delta T) \quad (7)$$

that aims to minimize the squared variation of the body vertical velocity.

5 EXPERIMENTAL RESULTS: ALGORITHM PARAMETERS

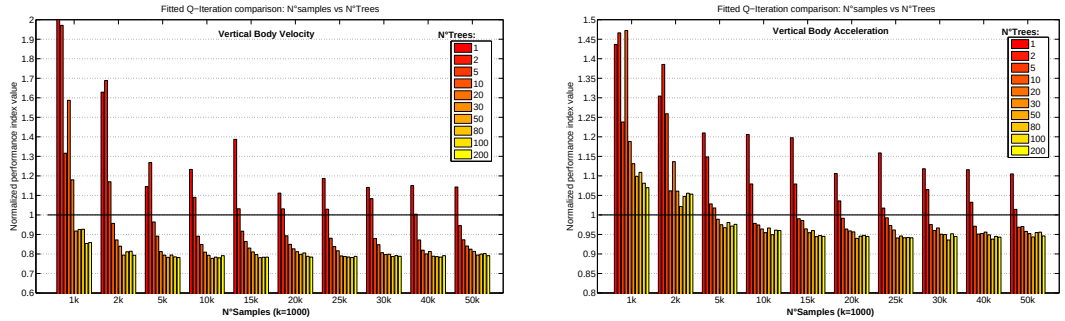
We performed a set of experiments in order to compare performance with different parameterizations. Samples are generated by controlling System (1) with a random policy and by feeding it with a road disturbance $z_r(t)$ designed as an integrated band-limited white noise. This signal is a realistic approximation of a road profile and excites all the system dynamics (Hrovat, 1997). The number of samples ranges from 1000 (10 seconds system simulation at 100Hz) to 50000 (500 seconds simulation).

The optimization horizon has been fix to 10 since it does not play a central role in this control problem. The number K of regressor's cut points is set to the dimension of the input space (in this case $K = |S| + |A| = 4$, see (Geurts et al., 2006)) . The number of trees depends mainly on the problem complexity and ranges from 1 to 100 in our experiments. Finally, the number of samples into a leaf, that affects the regressor's generalization ability, ranges from 2 to 1000. For each parameterization two cost functions have been evaluated: J_1 and J_2, which are obtained by learning the control policy with R_1 and R_2 respectively.

Figure 2 shows a comparison of policy obtained by varying both the number of samples and the number of samples in a leaf (NMin). Cost function values are normalized by Mixed SH-ADD policy. Results showed that as the number of samples increases, the cost decreases. Conversely, NMin and cost have a quadratic relationship. Lower values of NMin lead to over-fitting, larger values lead to a poor policy approximation, while intermediate values ($NMin = 50$) obtained the best performance.

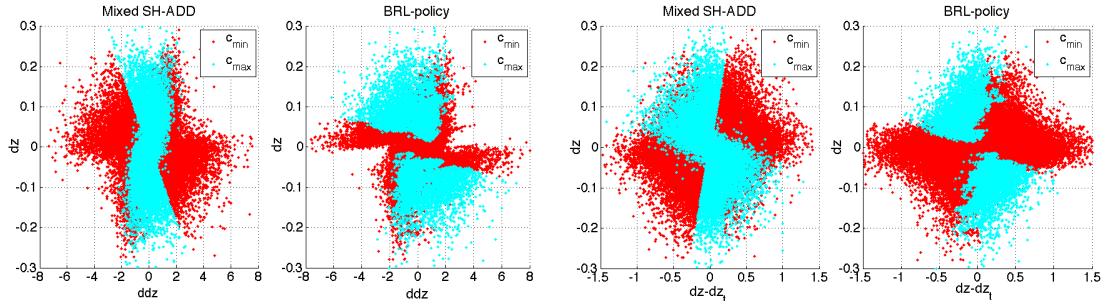
In Figure 3, a comparison of cost functions obtained by varying both the number of samples and the number of trees is presented. Again, index values are normalized by the Mixed policy. The performance improves by increasing both the number of samples and the number of trees. Nonetheless, after a certain value ($NTree >= 50$) no significant cost reduction is observed, while the computational cost grows linearly.

Figures 2 and 3 point out that the overall BRL-policy behavior is better than Mixed SH-ADD one on both indexes: J_1 and J_2. The more samples we have, the more accurate the learned policy will be. Few samples can be used, but choosing NMin or trees number can be critical.



(a) (b)

Figure 3: Comparison of vertical body velocity (J_2) and acceleration (J_1) indexes w.r.t number of samples and number of trees. J_1 and J_2 are normalized by Mixed SH-ADD policy ($J_1 = J_2 = 1$).



(a) (b)

Figure 4: Graphical representation of Mixed SH-ADD and BRL policies projected on $\ddot{z}(t)$, $\dot{z}(t)$ (a) and on $\dot{z}(t)$, $\dot{z}(t) - \dot{z}_t(t)$ (b).

6 EXPERIMENTAL RESULTS: POLICY COMPARISON

The experiments of Section 5 identified good parameters value for the estimation of an optimal policy using the BRL technique: state space $\mathcal{S}_3$, action space $\mathcal{A}$, reward function $\mathcal{R}_2(s(t),a(t))$ (Equation 7), 50K samples (500 seconds systems simulation), 10 fitted horizon, 50 tress, 5 random splits and $Nmin = 50$.

BRL-policy is a multi-dimensional control map that associates to every measurable state $< \ddot{z}(t), \dot{z}(t), \dot{z}(t) - \dot{z}_t(t) >$ a control action $c_{in}(t)$. A graphical representation of this map is depicted in Figure 4, where it is compared to the one obtained by controlling the semi-active system with the Mixed SH-ADD rule (quasi-optimal algorithm).

Figure 4 shows that BRL policy is very similar to the one associated to the Mixed SH-ADD algorithm. The BRL policy tends to prefer a high-damped suspension. The main differences between BRL map and Mixed SH-ADD can be highlighted around the origin of the axis. However, notice that, in such a situation, any selected damping has small influences on the body dynamics.

The performances of the semi-active suspension system fed with a random signal $z_r(t)$ and ruled by the BRL-policy has been evaluated in time and frequency domain. The frequency domain analysis in reported in Figure 5, which depicts the approximate

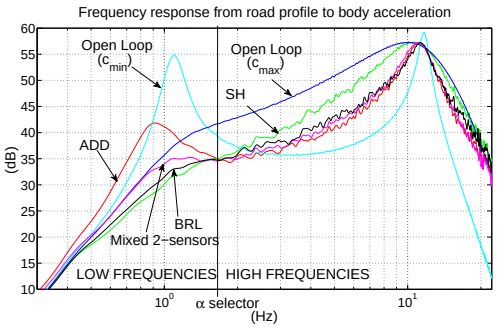


Figure 5: Frequency response from road profile to vertical acceleration of different policies: c_{max}, c_{min}, ADD,SH,Mixed SH-ADD and BRL-policy.

frequency response obtained as the ratio between the power spectrum of the output $\ddot{z}(t)$ and the input $z_r(t)$.

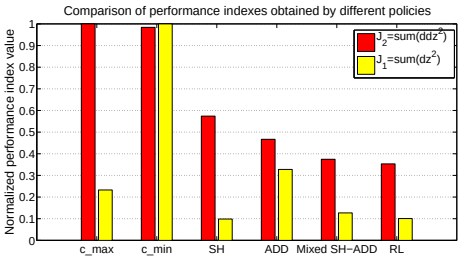


Figure 6: Comparison of different policies by using both cost function $J1$ and J_2.

The time domain results are condensed in Figure 6 where the cost functions J_1 and J_2 are reported for different control strategies and compared to the extreme passive configurations.

Figure 5 shows that BRL policy outperforms the Mixed SH-ADD at low frequency. This is paid in terms of filtering at high frequencies where Mixed SH-ADD shows a better behavior. Figure 5 points out that BRL policy provides the overall best performance in terms of minimization of integral of squared vertical body accelerations.

7 CONCLUSIONS

In this work we applied Batch Reinforcement Learning (BRL) to the design problem of optimal comfort-oriented semi-active suspension which has not been solved with standard techniques due to its complexity. Results showed that BRL policy provides the best results in terms of road disturbance filtering. However the achieved performances are not far from the ones obtained by the Mixed SH-ADD. Thus, comparing the numerical approximation given by BRL, against the analytical approximation given by the Mixed approach, we showed that they result in a similar strategy. This is an important finding which shows how numerical-based model-free algorithms can be used to solve complex control problems. Since BRL techniques can be applied to systems with unknown dynamics and are robust to noisy sensors, we expect to obtain even larger improvements on real motorbikes, as shown by preliminary experiments.

REFERENCES

Ahmadian, M., Reichert, B. A., and Song, X. (2001). System non-linearities induced by skyhook dampers. *Shock and Vibration*, 8(2):95–104.

Antos, A., Munos, R., and Szepesvari, C. (2008). Fitted q-iteration in continuous action-space mdps. In Platt, J., Koller, D., Singer, Y., and Roweis, S., editors, *Advances in Neural Information Processing Systems 20*, pages 9–16. MIT Press, Cambridge, MA.

Ernst, D., Geurts, P., Wehenkel, L., and Littman, L. (2005). Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6:503–556.

Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, 63(1):3–42.

Guardabassi, G. and Savaresi, S. (2001). Approximate linearization via feedback - an overview. *Survey paper on Automatica*, 27:1–15.

Hrovat, D. (1997). Survey of advanced suspension developments and related optimal control applications. *Automatica(Oxford)*, 33(10):1781–1817.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: a survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Karnopp, D. and Crosby, M. (1974). System for Controlling the Transmission of Energy Between Spaced Members. US Patent 3,807,678.

Riedmiller, M. (2005). Neural fitted q iteration - first experiences with a data efficient neural reinforcement learning method. In *ECML*, pages 317–328.

Sammier, D., Sename, O., and Dugard, L. (2003). Skyhook and H8 Control of Semi-active Suspensions: Some Practical Aspects. *Vehicle System Dynamics*, 39(4):279–308.

Savaresi, S., Silani, E., and Bittanti, S. (2005). Acceleration-Driven-Damper (ADD): An Optimal Control Algorithm For Comfort-Oriented Semiactive Suspensions. *Journal of Dynamic Systems, Measurement, and Control*, 127:218.

Savaresi, S. and Spelta, C. (2007). Mixed Sky-Hook and ADD: Approaching the Filtering Limits of a Semi-Active Suspension. *Journal of Dynamic Systems, Measurement, and Control*, 129:382.

Savaresi, S. and Spelta, C. (2008). A single-sensor control strategy for semi-active suspensions. *To Appear*, -:–.

Silani, E., Savaresi, S., Bittanti, S., Visconti, A., and Farachi, F. (2002). The Concept of Performance-Oriented Yaw-Control Systems: Vehicle Model and Analysis. *SAE Transactions, Journal of Passenger Cars - Mechanical Systems*, 111(6):1808–1818. ISBN No.0-7680-1290-2,.

Sutton, R. and Barto, A. (1998). *Reinforcement Learning: An Introduction*. MIT Press.

Valasek, M., Kortum, W., Sika, Z., Magdolen, L., and Vaculin, O. (1998). Development of semi-active road-friendly truck suspensions. *Control Engineering Practice*, 6:735–744.

Watkins, C. (1989). *Learining from Delayed Rewards*. PhD thesis, Cambridge University, Cambridge,England.

Williams, R. (1997). Automotive active suspensions Part 1: basic principles. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 211(6):415–426.

POSITION CONTROL OF A SERVO-PNEUMATIC SYSTEM

Hybrid Fuzzy P+I Controller of a Servo-Pneumatic Fatigue Simulator

Marco Santos, Jorge Ferreira
Department of Mechanical Enginnering, University of Aveiro, Campus Universitário de Santiago, Aveiro, Portugal
marco.santos@ua.pt, jaff@ua.pt

José Simões
Department of Mechanical Enginnering, University of Aveiro, Aveiro, Portugal
jose.simoes@ua.pt

Keywords: Servo-pneumatics, Fuzzy control, PI control, Hybrid fuzzy P+I control, High position accuracy, Biomechanical devices, Fatigue tests, Instrumented hip joint prosthesis.

Abstract: This paper proposes a hybrid fuzzy P+I controller for a servo-pneumatic machine to perform and monitor tests on biomechanical devices, such as orthopaedic prosthesis. The methodology followed is based upon the CompactRIO®, a real-time platform, and the system was fully programmed using LabVIEW language. Separate algorithms of a PI, proportional fuzzy and hybrid fuzzy P+I controllers were developed and compared. The performance of the overall system has already been tested and the experimental results for position control show that the PI controller can reach 2 μm of accuracy but with a very slow rise time. However, the same accuracy can be achieved with Hybrid Fuzzy P+I controller, although with a fast rise time and neglected overshoot. The authors can conclude that this proposal can successfully overcome unknown nonlinear parameters of the pneumatic system and has high position control accuracy.

1 INTRODUCTION

More than 250000 surgeries of total knee replacement and 180000 surgeries of total hip replacement are performed in the US every year. The smart hip joint prosthesis is a new research field, which is integrated in the Biomechanics Research Group (University of Aveiro) strategy. Total Hip Replacement (THR) arthroplasty is currently one of the most performed elective surgical procedures. The most serious complication of THR is loosening of the prosthetic stem and cup. No technique is capable of determining with exactness the levels of loosening, the reasons and the regions of the implant were it occurs with time. It has been referred that more than 80% of the non-successes are due to implant loosening. In this context, the PTDC/EME-PME/70824/2006 project, still running, has been financed, whose main aim is to develop a cemented and instrumented hip prosthesis with sensorial capacities to detect the degree of implant loosening and the regions where it occurs with time, through a non-invasive method that can be used to define clinical correction and prevention methodologies. Therefore, medical staff could access to "continuous" information about the evolution of the implant behaviour, providing means to avoid the presence of patients frequently in the medical office. The work here presented is mainly related with the project presented above and with an ongoing project where the principal aim is to develop a methodology to produce and study ultra high molecular weight polyethylene reinforced with carbon nanotubes (CNT/UHMWPE composites) and evaluate its suitability for enhancing the wear resistance of acetabular cups and therefore minimizing the above highlighted issues.

This background drew the need to develop a high position control and force accuracy of a 1 degree-of-freedom (DOF) servo-pneumatic machine developed by Biomechanics Research Group of the University of Aveiro (Santos, et al., 2008) to carry out the required fatigue simulation tests or any kind of pneumatic force up to 3 *kN*. The control system must also be prepared/easily fitted to answer to the accuracy requirements related with new researches on smart implants which may occur in the future. The fatigue simulator is shown in figure 1.

2 MECHANICAL APPARATUS

The main purpose of the fatigue simulator is the simulation of biomechanical actions to the static and dynamic characterization of synthetic femurs and tibias, with and without prostheses. This goal requires the tracking of positional reference trajectories through an upright course, as well as force references. An aluminum frame structure was reinforced to give more stability. The movement of the cylinder's piston and the load cell were protected with two lateral linear guides to compensate asymmetric loads. This press has already been used to perform long time running fatigue tests for more than 2 weeks running non-stop.

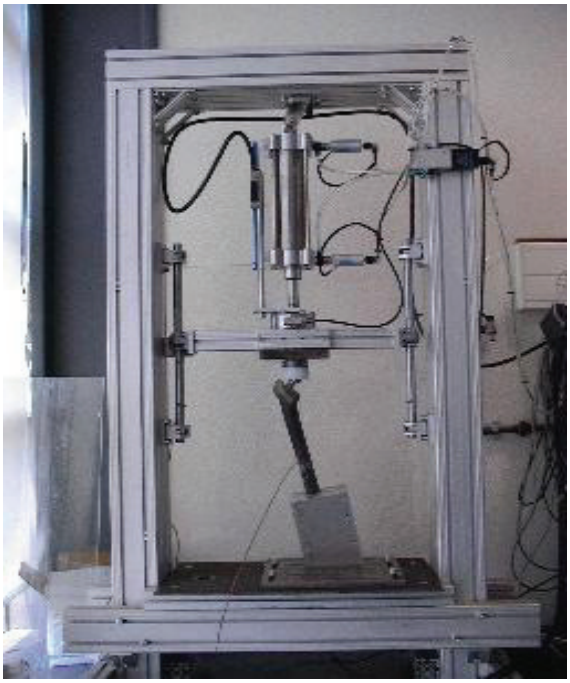


Figure 1: Mechanical Apparatus.

3 INSTRUMENTATION

The servo-pneumatic system consists of:
- A double effect pneumatic cylinder (Festo CRDNGS-80-200-PPV-A) with a 200 *mm* length and 80 *mm* diameter to ensure 3016 *N* at 6 *bar*;
- A servo-valve (Festo MPYE-5-1/8-HF-010-B) with 750 *l/min* capacity, to establish the amount of air circulating in each of the two actuator's chambers;
- An optical linear scale (Fagor SV- B220) with resolution of 1 μm is used to measure the machine's actuator moving mass.
- A load cell (AEP TC4), with 10 *kN* capacity and 0.1% resolution of that value, is used to measure the applied force to the biomechanical device.

4 HARDWARE PLATFORM

The National Instruments PAC CompactRIO® ensures the interface and the connection between the control software and all the instrumentation devices. It is constituted by a Reconfigurable CompactRIO® Intelligent Real-Time Controller/Web Server NI cRIO-9002, a four-slot reconfigurable embedded chassis NI cRIO-9103, a 16 bit Analog Input NI cRIO-9215, a Digital Input NI cRIO-9411 and a 16 bit Analog Output NI cRIO-9263. The real-time controller NI cRIO-9002 has its own 195 *MHz* industrial processor, 32 *MB* DRAM memory and a FTP server. With its embedded LabVIEW real-time ETS and a reconfigurable Field Programmable Gate Array (FPGA), this platform makes possible the development of high speed deterministic control applications with high flexibility.

5 SOFTWARE PLATFORM

All the control, monitor and data acquisition software were implemented using LabVIEW 8.0 Professional Development System and LabVIEW Reconfigurable Software Development Kit (includes LabVIEW Real-Time 8.0 and LabVIEW FPGA 8.0 modules), to ensure the controller autonomy, the easy interface with the user operator, the possibility of remote web monitoring and the easy upgrade of controllers. The hardware platform allows a three layer distributed software, as shown in figure 2. Previous work on the design of position and force controllers using this platform were conducted by Santos *et al.* (Santos, *et al*., 2008).

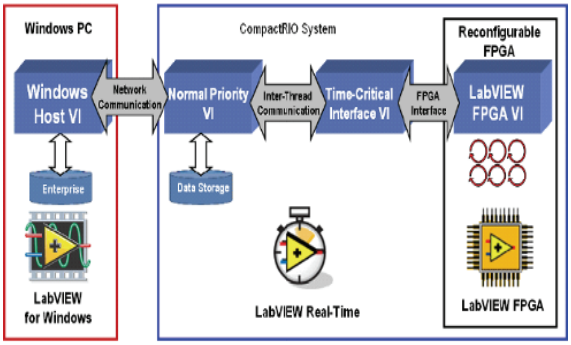


Figure 2: Three layer distributed software.

6 THE PNEUMATIC SYSTEM

The pneumatic servosystem represented in figure 3, is composed by a pneumatic cylinder, which performs movement, and a 5/3 servovalve, that modulates the amount of air entering the cylinder. P_i, T_i and A_i represent pressure, temperature and piston areas of chambers *i*. *M* is the actuator moving mass and *x* is its position. The air supply pressure P_s is set at 6 *bar*. The servovalve and the pneumatic cylinder are the two system's main blocks. Pneumatic systems usually present a set on nonlinearities that creates problems to close loop controllers. When high accuracy positioning tasks is requested, electrical solutions are normally chosen instead of servopneumatic solutions. The complexity in controlling servopneumatic systems is mainly due to the air compressibility, the piston friction and the non-linear behaviour of the servovalve. In the present case, the servovalve is the highly nonlinear element of the servopneumatic system. The assumption that the orifices area of the servovalve varies linearly with the command input can lead to large modelling errors near the spool central position. Furthermore, there is also the temperature and pressure dynamics of the actuator chambers that should not be neglected (Carneiro, 2006).

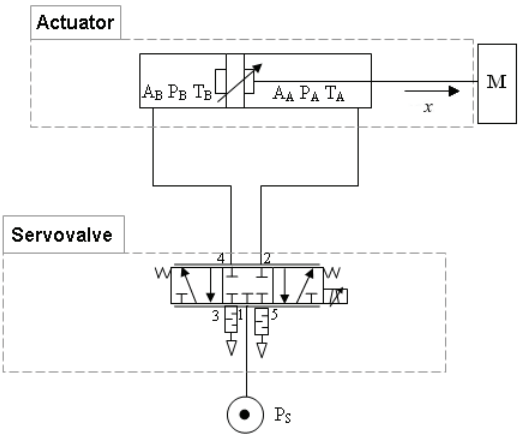


Figure 3: Pneumatic Servosystem.

7 THE CONTROL SYSTEM

The general positioning accuracy in pneumatic control systems only can reach to the range from ±0.1 *mm* to ±0.05 *mm*. Xiang and Wikander (Xiang, 2004) and Carneiro (Carneiro, 2007) have achieved positioning errors below 5 *μm,* through the application of nonlinear control methodologies based in the information from mathematical models of the system, respectively feedback linearization and sliding mode control, to deal with the nonlinearities of the pneumatic system. However, they also have to deal with the inaccuracy of the mathematical model, because it cannot perfectly represent all possible dynamics of the physical process. So, the control system performance depends on the mathematical model accuracy. And cannot be forgotten that the application of some nonlinear control techniques requires lower-order "design models". To deal with these problems, Pai and Shih (Pai, *et al*., 2003) have developed a fuzzy PD controller and they've got a positioning accuracy of 20 *nm* (equal to the resolution of the linear digital scale), the best experimental result found in the literature by the authors. They used heuristic information to build a "human-in-the-loop" controller and have written down a set of rules on how to control the process. Then, they incorporated them into a fuzzy controller to emulate their decision-making process. They have defined the error and change of error as inputs, one output and only nine rules; mamdani control rules and the maximum-minimum algorithm; and "center of the gravity" defuzzification method. They don't detail neither about the linear digital scale nor the 3/5-port proportional control valve. The control system was only tested with step and small multi-step inputs.

As opposed to "conventional" control approaches, where the focus is on modelling and the use of the model to build a controller, the fuzzy control is concerned about the intuitive understanding of how to best control the process (Passino, et al., 1998).

The performances of a PI and fuzzy logic closed loop control strategies have been studied. Finally, a hybrid fuzzy-P+I controller was implemented to take the advantages of both (Liu, et al, 2007).

7.1 PI Control

The development of a PI controller requires finding the appropriate proportional and integral parameters K_P and K_i, respectively, as shows in Equation 1.

$$u(t) = K_P e(t) + K_i \int_0^t e(t)dt \qquad (1)$$

Because of the limited displacement of the pneumatic and the related integral component evolution, an anti-windup technique is used to introduce a dead zone between non-saturated output values (Carneiro, 2007). The optimized parameters found were K_p=*100* and K_i=*10*.

7.2 Fuzzy Control

The Fuzzy Logic Control (FLC) is based on an input-output function that maps each numerical input to a low-resolution quantization interval and calculates the control signal based on an output quantization interval (Ferreira, *et al.*, 2006). The input values for the position fuzzy feedback control were the position error. Seven membership functions were defined for the error and the same number to the output signal, as shown in the program code below (transcribed from *.fis* file) and in figures 4 and 5. Table 1 shows the base-rule. Mamdani control rules and maximum aggregation method are applied. The rule's weight was set equal to 1. The "center of the gravity" method is used to defuzzify and to get the accurate control signal.

```
[Input1] Name='error'
Range= [-1 1]
MF1='-3': [-1 -1 -0.4 -0.1]
MF2='-2': [-0.2 -0.1 0]
MF3='-1': [-0.02 -0.01 0]
MF4='0':  [0 0 0]
MF5='1':  [0 0.01 0.02]
MF6='2':  [0 0.1 0.2]
MF7='3':  [0.1 0.4 1 1]

[Output1] Name='signal'
Range= [-1 1];
MF1='-3': [-1 -1 -0.5 -0.25]
MF2='-2': [-0.4 -0.25 -0.05]
MF3='-1': [-0.1 -0.05 0]
MF4='0':  [0 0 0]
MF5='1':  [0 0.05 0.1]
MF6='2':  [0.05 0.25 0.4]
MF7='3':  [0.25 0.5 1 1]
```

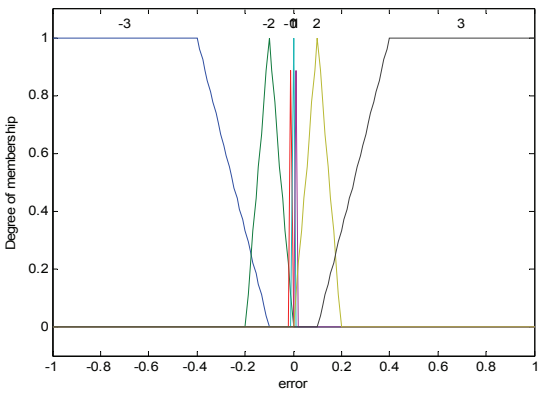


Figure 4: Membership function of the position error.

The saturation limits of the position error were set to -200 and 200 *mm*. The accuracy of the approximation depends mostly on the membership functions and the rules.

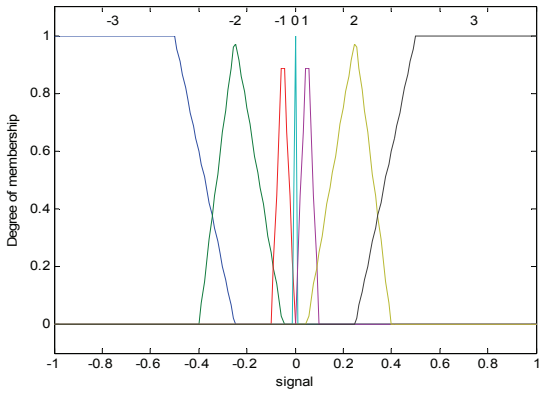


Figure 5: Membership function of the control output.

Table 1: Fuzzy Rule matrix.

	Error						
	-3	-2	-1	0	1	2	3
Signal	-3	-2	-1	0	1	2	3

The optimized control system response can also be seen as a non-linear operation function shown in Figure 6. It models the servovalve opening/closing rate μ_{FUZZY} as a function of the position error. This design has the behaviour of a *fuzzy proportional* (FP). It is the simplest "human-in-the-loop" controller, although must link the requirements of high accuracy, fast rise time and neglected overshoot results with an easy control system parameterization.

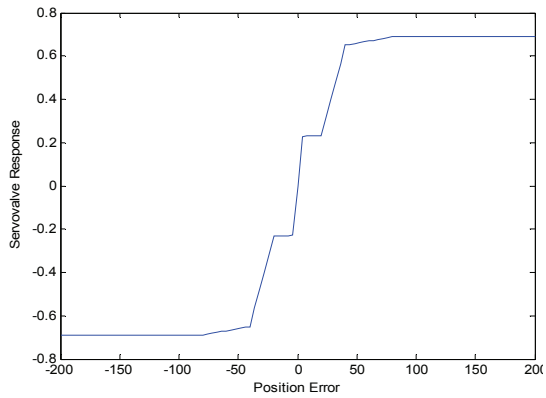


Figure 6: Operation function of the FP controller.

Many researchers have been studying the architecture of fuzzy controllers. On-line processing of the fuzzy inference rules requires high computational processing cost. Several authors proposed a controller in which the values of the membership functions must be fixed before computation of the control process and showed that the fuzzy algorithm can be converted into matrices to represent the parameterized fuzzy model. One of

the advantages of a matrix representation is that its computation is faster than the fuzzy control statements. To carry out a real-time computing system with CompactRIO® Controller using the fuzzy logic controller, a 1D *Look-up Table* is generated from the Fuzzy Logic toolbox in Matlab. Once a value for the position error is found, it is cross-referenced in the *Look-Up Table* to find the control output. The selected parameters which create the table were intervals of 0.02 for the position error. For inputs values between these intervals, the output value is established by linear interpolation. So, this algorithm is run *off-line* and the generated table is sent to cRIO-9002 though FTP protocol. The closed loop control was design into the Real-Time Controller. The LabVIEW block diagram is shown in Figure 7. The optimized parameters were $K_e=1$ and $K_s=1$.

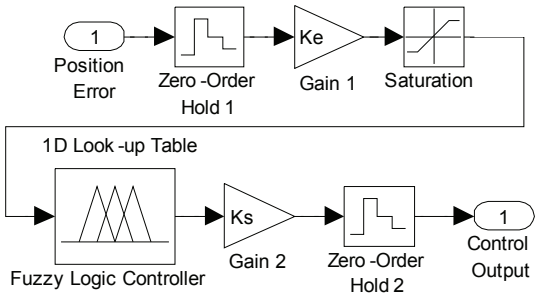


Figure 7: LabVIEW block Diagram of the Fuzzy Logic Controller.

7.3 Hybrid Fuzzy P+I Control

Fuzzy controllers do not need precise information about the nonlinearities of the system to be effective. However, it is useful the use of the integral of the error to deal with steady-state error of the position variable, which is difficult to eliminate only with the developed FLC. Hence, it was developed a hybrid system where it was given a weight of 91,5% to the fuzzy logic controller and a weight of 8,5% to the integrator contribution, in order to achieve its optimized performance. Equation 2 shows the hybrid Fuzzy P+I control law.

$$u(t) = 0.915\mu_{FUZZY} + 0.085K_i\int_0^t e(t)dt \quad (2)$$

The optimized value of K_i is *10*. An anti-windup technique is also used.

8 EXPERIMENTAL RESULTS

All the control algorithms presented in section 7 were built-in into cRIO-9002 Controller and compared. Several step signals were applied as the position reference in the control experiments and the cylinder position measurements were recorded. Figures 8 and 9 show the system response when each of the control methods is applied with a step signal reference from 0 to 100 *mm*.

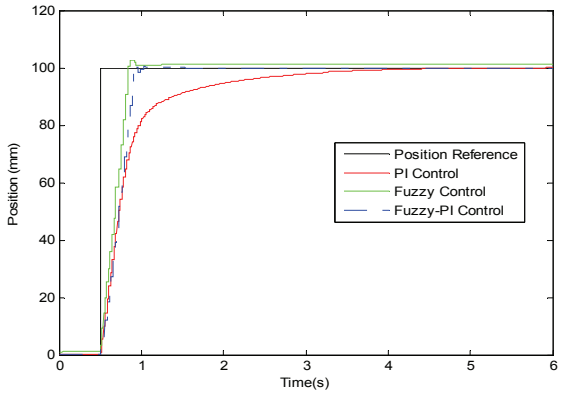


Figure 8: Step signal response curves of PI, Fuzzy and Hybrid Fuzzy-P+I Controllers.

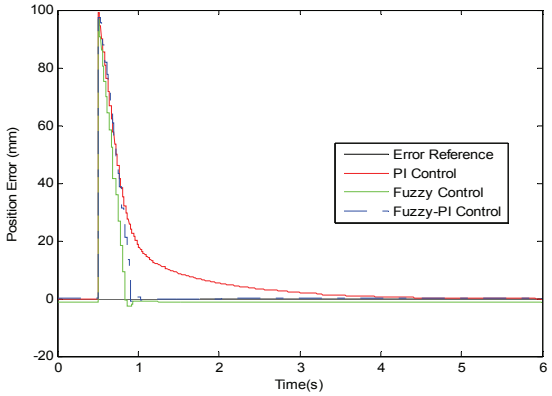


Figure 9: Step signal error curves of PI, Fuzzy and Hybrid Fuzzy-P+I Controllers.

The steady-state position error between actual position and the set point value is 1,104 *mm* for FP controller and 2 μm for Hybrid Fuzzy-P+I controller. With the PI controller, a position error of 0,014 *mm* can be achieved only after 60 seconds. With the Hybrid Fuzzy-P+I the position error is less than 0,5 *mm* after 0,53 seconds response, 0,05 *mm* after 0,95 seconds, and 2 μm steady-state position error after 1,015 seconds response. The overshoot was 0,358 *mm*. When the application of the Hybrid Fuzzy-P+I

controller makes the mass *M* to reach its steady-state position, the position error with the PI controller is about 8,75 *mm*.

It was also carry out other experiences to study the accuracy and repeatability of the overall control system. Results are shown in Figures 10, which are relative to the steady-state position error of the PI and Hybrid Fuzzy-P+I control, respectively, when were applied step signal references from 0 to REF_signal *mm* (multiples of 5 *mm*, i.e., steps: 0 – 5 *mm*; 0 – 10 *mm*; …; 0 – 175 *mm*; 0 – 180 *mm*).

The worst steady-state position error of the PI Control is 0,084 *mm*, but 3 times reaches below 2 μm of accuracy and 13 times reaches a accuracy below 10 μm, although its average is 11 μm and needs a long amount of time to reach these accuracies. PI control can accurately control the position of a pneumatic system, however it takes too long. The sticking and restarting phenomena become more evident when applied a step signal reference from 0 to REF_signal *mm* where 80 *mm* ≤ REF_signal ≤ 140 *mm*.

The worst steady-state position error of the Hybrid Fuzzy-P+I Control is 0,04 *mm*, but 7 times reaches below 2 μm of accuracy and 18 times reaches a accuracy below 5 μm, although its average is 8,6 μm. This controller has achieved a diminution in the overshoot and in the steady state error compared with PI and FP controllers, although it suffers some late correcting the position error. The FP controller has the shortest rise time and reaches early the steady-state position. With FLC, with or without the hybrid solution, the sticking and restarting phenomena was not observed.

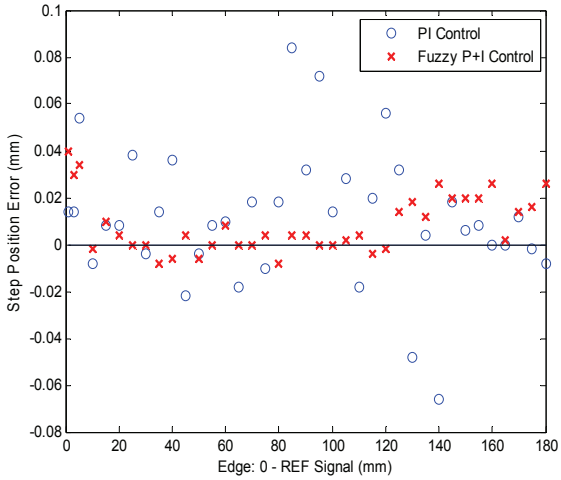


Figure 10: Steady-state position error of PI and Hybrid Fuzzy-P+I Control when step references from 0 to REF_signal *mm* were applied.

9 CONCLUSIONS

The present paper describes the design and control of a pneumatic press to perform controlled tracking of positional reference trajectories. The control system of the servo pneumatic machine was implemented with LabVIEW using CompactRIO® hardware. Conventional PI, proportional Fuzzy and Hybrid Fuzzy-P+I control strategies were compared in the position control of a moving mass over 200 *mm* course. The Hybrid Fuzzy-P+I Controller provides the best performance for the performed position control experiments: short rise time, small overshoot and a steady-state position error that can reach the encoder resolution, less than 2 μm. It has particularly advantageous in terms of simplicity of design and implementation. There is an ongoing work to implement a high performance force controller using FLC and the overall pneumatic system simulation with Matlab/Simulink® platform.

REFERENCES

Carneiro, J., Almeida, F., 2006. *Modeling Pneumatic Servovalves using Neural Networks*, Proceedings of the 2006 IEEE Conference on Computer Aided Control Systems Design, pp. 790-795, Munich, Germany.

Carneiro, J., 2007. *Modelação e Controlo de Actuadores Pneumáticos Utilizando Redes Neuronais Artificiais*, PhD Thesis in Mechanical Engineering, University of Porto, Portugal.

Ferreira, J., Sun, P. Grácio, J., 2006. *Design and Control of a Hydraulic Press*, Proceedings of the 2006 IEEE Conference on Computer Aided Control Systems Design, pp. 814- 819, Munich, Germany.

Liu, H., Lee, J., Li, B., 2007. *High Precision Pressure Control of a Pneumatic Chamber Using a Hybrid Fuzzy PID Controller*, International Journal of Precision Engineering and Manufacturing, Vol. 8, No. 3, pp. 8-13.

Pai, K., Shih, M., 2003. *Nanoaccuracy Position Control of a Pneumatic Cylinder Driven Table*, JSME Int'l Journal, Series C, Vol. 46, No. 3, pp. 1062-1067.

Passino, K., Yurkovich, S., 1998. *Fuzzy Control*, Addison Wesley Longman, Menlo Park, CA.

Santos, M., Talaia, P., Ramos, A., Ferreira, J., Oliveira, M., 2008. *Servo-Pneumatic Machine to Perform and Monitor Tests on Biomechanical Devices*, Controlo 2008: Proceedings of the 8th Portuguese International Conference on Automatic Control, pp. 784 – 789, Vila Real, Portugal.

Xiang, F., Wikander, J.,2004..*Block-oriented Approximate Feedback Linearization for Control of Pneumatic Actuator System*, Control Engineering Practice 12(4), pp. 6113-6119.

A FUZZY-CONTROLLED INFLUENCE FUNCTION FOR THE CULTURAL ALGORITHM WITH EVOLUTIONARY PROGRAMMING APPLIED TO REAL-VALUED FUNCTION OPTIMIZATION

Intelligent Control Systems and Optimization

Mário Augusto Torres, Otávio Noura Teixeira and Roberto Limão de Oliveira
Universidade Federal do Pará, Belém, PA, Brazil
mario.act@gmail.com, onoura@gmail.com, limao@ufpa.br

Keywords: Cultural Algorithms, Evolutionary Programming, Fuzzy Inference Systems, Real-valued Unconstrained Function Optimization.

Abstract: In this paper, we propose a fuzzy system to act as a control mechanism for the evolutionary process of search of a Cultural Algorithm with Evolutionary Programming (CAEP) applied to real-valued function optimization. The fuzzy system uses population knowledge to adjust the Influence Factor that represents the intensity of the influence of the Variation operator of the CAEP model, therefore adjusting the search process. This paper also presents a comparative analysis of the proposed influence function using well-known benchmarking functions.

1 INTRODUCTION

Fuzzy Systems have been used as control mechanisms in many applications. From the control of industrial processes to self adapting air-conditioners, fuzzy control systems have been successfully employed due to their capability of processing uncertain, imprecise knowledge.

Cultural Algorithms (CA) are a class of evolutionary computational models proposed by Reynolds, derived from observing the cultural evolution process in nature (Reynolds, 1994). CA categorizes the population experience in several knowledge sources stored in a belief space and utilizes this knowledge to guide the further evolution of the population.

The use of fuzzy reasoning as a controller of the process of acquiring experimental knowledge was proven to be successful in increasing the performance of a cultural algorithm with evolutionary programming (CAEP) system (Chung, 1997).

The fully-fuzzy Cultural Algorithms framework approach managed to obtain even better results in 12 of the functions that the crisp version of the framework could not always provide the solution in the allotted number of generations (Zhu, 1998). The fully fuzzified approach for the Cultural Algorithm with Evolutionary Programming (CAEP) system consisted of a fuzzy acceptance function, a fuzzy representation of the knowledge contained within the belief space and a fuzzy influence function (Zhu, 1998).

Still, we believe there is a chance for further improvement in the fuzzy influence function proposed in (Zhu, 1998), as a control mechanism for the search process. The proposal of this paper utilizes a fuzzy inference system to regulate the intensity of the EP variation operator based on imprecise search optimization knowledge, more specifically cultural influence level knowledge.

2 CULTURAL ALGORITHMS

As stated above, Cultural Algorithms are a class of evolutionary computational models proposed by Reynolds, derived from observing the cultural evolution process in nature (Reynolds, 1994). CA has three major components: a Population Space, a Belief Space and a Communication Protocol that determines how knowledge is exchanged between

the first two components.

The population space can support any population-based computational model, such as Genetic Algorithms and Evolutionary Programming (Reynolds et al., 2005). The belief space is a knowledge repository, gathered from the behaviour and individual experiences of the members of the population space. Saleem (Saleem, 2001) defines five different knowledge sources, stored and manipulated within the belief space: **Situational Knowledge,** exemplars of successful and unsuccessful behaviours in the population space; **Normative Knowledge,** defining the range of acceptable or desirable behaviours; **Domain Knowledge**, such as knowledge about domain objects, their properties and relationships; **History Knowledge**, that stores temporal patterns of behaviour; and **Topographical Knowledge**, that stores spatial patterns of behaviour of the search space.

The communication protocol defines how the members of the population space contribute to the knowledge gathering within the belief space and how the knowledge stored in the belief space influences the individuals in the population space. To achieve this, two distinct channels are defined: the **Acceptance Function** selects the individuals whose behaviours and experiences will contribute to update the knowledge in the belief space; and the **Influence Function** defines how the knowledge stored in the belief space influences the operators that modify the individuals in the population space. Chung (Chung, 1997), Zhu (Zhu, 1998) and Rodrigues (Rodrigues, 2007) state that the influence function works as a self-adaptation mechanism for the evolutionary process, for it adapts the population operators according to the gathered knowledge.

The Cultural Algorithm, therefore, is a dual inheritance system that characterizes evolution in human culture at both the macro-evolutionary level, which takes place within the belief space, and at the micro-evolutionary level, which occurs in the population space (Reynolds et al., 2005).

Figure 1 depicts the main components of the Cultural Algorithms framework and their relationships, adapted from (Saleem, 2000).

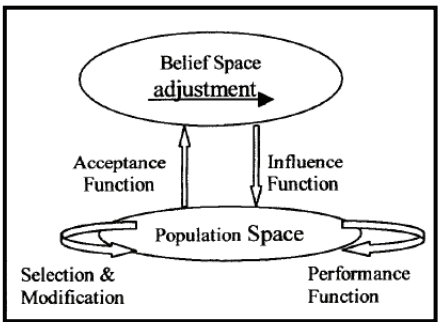


Figure 1: Cultural Algorithms Framework (Saleem, 2000).

3 THE CAEP FRAMEWORK

The CAEP (Cultural Algorithm with Evolutionary Programming), as defined by Chung (Chung, 1997), is a Cultural Algorithm framework with Evolutionary Programming as its population component, and the global knowledge that is learned by the population expressed as Normative and Situational knowledge sources. It was successfully used by Chung in real-valued function optimization. In the following subsections, the CAEP framework is briefly explained.

3.1 Evolutionary Programming

Evolutionary Programming (EP) can usually be described, as in (Fogel, 1996):

$$x^{t+1} = s(v(x^t)) \quad (1)$$

where x^t is the population of solutions in the iteration t, $v()$ is the variation operator used to generate new solutions and $s()$ is the selection operator that determines which candidate solutions will survive to the next population x^{t+1}.

3.2 Belief Space Structure

The formal definition of the belief space in the CAEP framework is $< E, N_{[1,\ldots,n]} >$, where E is the set of exemplars of desirable behaviour and represents the situational knowledge. $N_{[1,\ldots,n]}$ is normative knowledge component, which consists of a set of interval information for each n parameter. Each interval in the N set is denoted as $< I, U, L >$, where I denotes a closed interval of real numbers x, represented as:

$$I = [l,u] = \{x \mid l \leq x \leq u\} \quad (2)$$

where l (lower bound) and u (upper bound) are initialized as the domain values. L_j represents the

performance score for the lower bound *l* for the parameter *j* and *Uj* represents the performance score for the upper bound *u* for the parameter *j* (Chung, 1997).

3.3 Acceptance Function

The acceptance function selects the individuals that will contribute with the formation of the knowledge in the belief space. There are many possible classes of acceptance functions. Chung (Chung, 1997) described a few of these functions.

The acceptance function used in the tests of the influence function described in this paper is the **Top 20%**. This function is static in nature and consists in selecting the top 20% of the individuals in the population space. It was chosen for its simplicity and because it provided one of the best results in Chung's tests (Chung, 1997).

3.4 Adjusting the Belief Space

In the belief space, the situational knowledge consists of the current and previous best individuals found so far. Formally, it is represented as $< \vec{E}^t, \vec{E}^{t-1} >$ and is adjusted by the following rule:

$$\vec{E}^{t+1} = \begin{cases} \vec{x}_{best}^t, & if\ f(\vec{x}_{best}^t) < f(\vec{E}^t) \\ \vec{E}^t, & otherwise \end{cases} \quad (3)$$

where $\vec{x}_{best}^t$ is the best individual (solution parameter vector) found in the population time *t* (Chung, 1997).

The normative knowledge component, *N*, is updated using the individuals selected by the acceptance function, which are used to calculate the current acceptable interval for each of the parameters of the individuals. In the following, *i* represents the individual with the lowest value for parameter *j* and *k* denotes the individual with the highest value for parameter *j*. The update rules for the left boundary and its fitness score for parameter *j* are:

$$l_j^{t+1} = \begin{cases} x_{i.j}^t, & if\ x_{i.j}^t \le\ l_j^t\ or\ f(x_i^t) < L_j^t) \\ l_j^t, & otherwise \end{cases} \quad (4)$$

$$L_j^{t+1} = \begin{cases} f(x_i), & if\ x_{i.j}^t \le\ l_j^t\ or\ f(x_i^t) < L_j^t) \\ L_j^t, & otherwise \end{cases} \quad (5)$$

where l_j^t denotes the lower limit of the acceptable interval for parameter *j* at generation (iteration) *t* and L_j^t represents the performance score for it. The update rules for the right boundary and its fitness score for parameter *j* are:

$$u_j^{t+1} = \begin{cases} x_{k.j}^t, & if\ x_{k.j}^t \ge\ u_j^t\ or\ f(x_i^t) < U_j^t) \\ u_j^t, & otherwise \end{cases} \quad (6)$$

$$U_j^{t+1} = \begin{cases} f(x_i), if\ x_{k.j}^t \ge u_j^t\ or\ f(x_i^t) < U_j^t) \\ U_j^t, & otherwise \end{cases} \quad (7)$$

where u_j^t denotes the upper limit of the acceptable interval for parameter *j* at generation (iteration) *t* and U_j^t represents the performance score for it.

3.5 Cultured EP Algorithm

The following pseudo-code was proposed by Chung (Chung, 1997) for a basic "cultured" EP algorithm and constitutes the skeleton algorithm for the CAEP framework. Steps (3) and (8), shown in bold characters, are the procedures added in order to introduce the cultural aspect in the EP algorithm. Note that step (4) represents the step where the influence function is applied for the CAEP framework and is where the self-adaptation occurs (Chung, 1997).

(1) Generate an initial population of *p* candidate solutions from an uniform distribution within the given domain for each parameter from 1 to *n*;

(2) Assess the performance score for each parent solution using the objective function;

(3) Initialize the belief space with the given problem domain and candidate solutions;

(4) Generate *p* new offspring by applying the variation operator, *v*(), as modified by the influence function. Now, there are *2p* solutions in the population;

(5) Assess the performance score for each offspring using the given objective function *f*;

(6) For each individual, select *c* competitors at random from the population of *2p* size. Next, conduct pair-wise competitions between the individual and the competitors and count the number of wins w_i for that individual;

(7) Select the *p* solutions with the greatest number of wins (w_i) to be the parents of the next generation;

(8) Update the belief space by accepting individuals using the acceptance function described in 3.3.

The belief space is adjusted according to the rules presented in 3.4.

(9) The process returns to step 4 unless the available execution time is exhausted or an acceptable solution has been found.

3.6 Chung's Influence Functions

The knowledge stored in the belief space can influence the evolutionary variation operator v in two ways: (1) determining the size of the mutation change, called step size, and (2) defining the direction of the variation, positive or negative (Chung, 1997). Chung proposed three different influence functions: The CAEP(Ns), CAEP(Ns+Sd), and the CAEP(Nsd). Chung showed that the CAEP(Nsd) had the best results. Thus, this influence function is described in the following.

3.6.1 CAEP(Nsd)

This version utilizes the normative knowledge to determine both the size and the direction of the variation. The basic idea is to perturb small in a random direction if an individual's parameter value is in the acceptable range; otherwise, perturb the parameter value towards the left or right boundary of the acceptable range for that parameter in the belief space. For all individuals $i = 1…p$ and parameters $j = 1…n$:

$$x_{p+i,j} = \begin{cases} x_{i,j} + \left|size(I_j) * N_{i,j}(0,1)\right|, if\ x_{i,j} < l_j^t \\ x_{i,j} - \left|size(I_j) * N_{i,j}(0,1)\right|, if\ x_{i,j} > u_j^t \\ x_{i,j} + \beta * sizeI_j * N_{i,j}(0,1), otherwise \end{cases} \quad (8)$$

where l_j^t and u_j^t represent the lower limit and upper limit for the parameter j in the generation t, respectively. β is set to 0.2.

4 THE FUZZY INFERENCE INFLUENCE FUNCTION (FIS-NSD)

Many works have been able to achieve some improvements in real-valued function optimization by making some aspects of Cultural Algorithms fuzzy. Chung (Chung, 1997) proposed a fuzzy acceptance function, based on a fuzzy inference engine to determine the percentage of accepted individuals in each generation, taking in consideration the current generation and the success ratio of the algorithm as the input of the engine, and was able to improve the overall performance of the algorithm in 34 benchmark functions; Zhu (Zhu, 1998) proposed a fully fuzzy cultural algorithm and was able to improve the results in Chung on 12 benchmark functions.

We propose an influence function based on those proposed by Chung and Zhu, and incorporating a fuzzy inference engine to better represent imprecise search optimization knowledge, more specifically cultural influence level knowledge.

We used the influence function Nsd proposed by Chung as the base mechanism to influence the variation operator. The following is the rule that defines the proposed influence function. For all individuals $i = 1…p$ and parameters $j = 1…n$:

$$x_{p+i,j} = \begin{cases} x_{i,j} + \boldsymbol{\omega_i} * \left|size(I_j) * N_{i,j}(0,1)\right|, if\ x_{i,j} < l_j^t \\ x_{i,j} - \boldsymbol{\omega_i} * \left|size(I_j) * N_{i,j}(0,1)\right|, if\ x_{i,j} > u_j^t \\ x_{i,j} + \boldsymbol{\omega_i} * \beta * size(I_j) * N_{i,j}(0,1), otherwise \end{cases} \quad (9)$$

where $\boldsymbol{\omega}_i$ represents the influence factor that modifies the intensity in which the variation operator is applied to the ith individual. This influence factor, similar to the step adjustment coefficient described by Zhu (Zhu, 1998), is designed to adjust the search process in a search optimization knowledge-based heuristic.

As stated in (Chung, 1997), the age of an individual is important information because if an individual is old, that means the it might be trapped in a local optimum. So, in order to escape, a larger perturbation might be necessary.

Another important parameter to be considered is the performance evaluation of an individual. If its fitness evaluation is considered to be poor, then it could mean that the individual is farther from finding the global optimum than the best individuals in the population, so it might be necessary that the change we apply in this individual is greater than that we apply in the best ones. The fitness evaluation rule of an individual is defined as a real value between 0 and 1.

The main idea is to regulate the intensity of the change in the variation operator applied to a parent individual using a fuzzy inference system. The fuzzy inference engine receives as input two variables, corresponding to the age (in number of generations) of the individual and the fitness evaluation of the individual, to determine the influence factor $\boldsymbol{\omega}_i$ for each individual i, from $i = 1…p$. The fuzzy inference system is shown in figure 2.

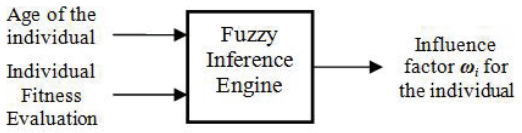


Figure 2: Fuzzy Inference System used to determine ω_i.

The antecedent membership functions are linear functions for the fuzzification process and are shown in the figures 3 and 4. In the fuzzy inference system, a set of input parameters, representing the age of the individual and its fitness evaluation, are mapped into one or more degrees of membership, e.g. Young, Adult and Old; Poor, Average and Good.

After the fuzzification process, the engine makes use of the rules shown in figure 4 to infer the degree of membership of the fuzzy output and provide a real-valued output ω_i. The membership function for the output variable InfluenceFunction is shown in figure 5.

The basic knowledge represented in the rules designed in the fuzzy inference system is the following: if the individual is Old or its fitness evaluation is Poor, then the influence factor applied in the variation operator for that individual is High; if the individual is Young or its fitness evaluation is Good, then the influence factor concerning this individual is Low.

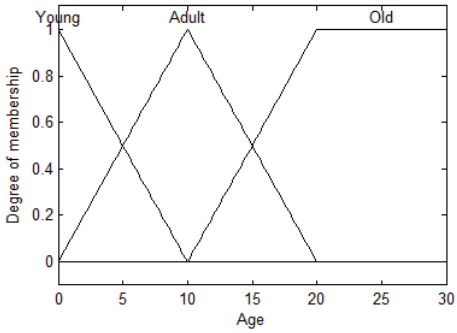


Figure 3: Membership Function for the Age parameter.

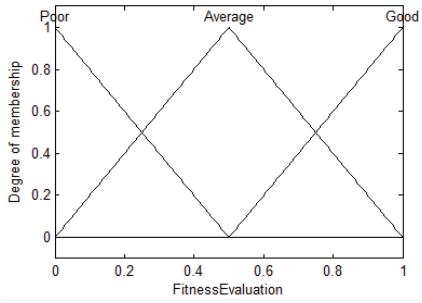


Figure 4: Membership Function for the Fitness Evaluation parameter.

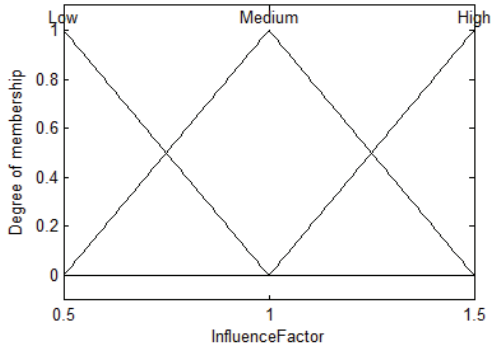


Figure 5: Membership Function for the output parameter, Influence Factor ω_i.

The fuzzy inference rule base used in the fuzzy inference system is shown in table 1.

Table 1: The Fuzzy Inference Rules used in the FIS.

	Poor	**Average**	**Good**
Young	Medium	Medium	Low
Adult	High	Medium	Medium
Old	High	High	Medium

5 TESTS DESCRIPTION

The approach was tested using a set of 14 of the well-known 25 CEC '05 benchmarking functions (Suganthan et al., 2005), both unimodal (F01 to F05) and multimodal - basic (F06 to F12) and expanded (F13 and F14). All functions were used with dimensionality n = 30.

According to Chung's results, the best influence function is the CAEP(Nsd). So, we used this CAEP configuration to compare with the proposed CAEP(FIS-Nsd) influence function. Both CAEP configurations use the top-20% as the acceptance function, a population size of 60 individuals, iterated tournament as the selection operator, and were executed for 25 runs, each run set to 300000 function evaluations (FEs) at maximum, equivalent to 5000 generations.

6 TESTS RESULTS

The results are shown in tables 2 and 3, depicting the minimum number of FEs used to solve the function, the average number of FEs required, the average fitness value of the best solutions and the success rate, for each CAEP configuration.

Table 2: Results for the 14 Functions for the CAEP(Nsd) configuration.

Function	MIN FEs	AVG FEs	BEST Fitness	AVG Fitness	Success %
F01	12360	12712.8	-450	-450	100
F02	155340	139003.2	-450	-450	100
F03	300000	300000	4.48E5	2.29E6	0
F04	300000	300000	-449.9996	-449.96	0
F05	300000	300000	1553.32	2315.67	0
F06	300000	300000	390.00	410.95	0
F07	300000	300000	4516.28	4516.28	0
F08	300000	300000	-119.14	-119.06	0
F09	300000	300000	-319.05	-260.31	0
F10	300000	300000	-173.88	-167.66	0
F11	300000	300000	95.47	117.13	0
F12	300000	300000	2.75E5	3.82E6	0
F13	300000	300000	-116.52	-115.85	0
F14	300000	300000	-287.01	-286.91	0

Table 3. Results for the 14 Functions for the CAEP(FIS-Nsd) configuration.

Function	MIN FEs	AVG FEs	BEST Fitness	AVG Fitness	Success %
F01	10320	10886.4	-450	-450	100
F02	151020	162852	-450	-450	100
F03	300000	300000	3.26E5	4.45E5	0
F04	268268	294892,8	-450	-449.996	36
F05	300000	300000	1964.69	2412.47	0
F06	300000	300000	390.34	397.56	0
F07	300000	300000	4516.28	4516.28	0
F08	300000	300000	-119.29	-119.10	0
F09	300000	300000	212.17	-186.38	0
F10	300000	300000	-173.91	-159.71	0
F11	300000	300000	128.62	129.48	0
F12	300000	300000	1562.40	14900.4	0
F13	300000	300000	-116.78	-116.42	0
F14	300000	300000	-286.68	-286.64	0

7 FINAL REMARKS

We observed that the addition of a fuzzy inference system to regulate the intensity of the influence function applied to the individuals alone can improve the performance of the CAEP(Nsd) configuration. However, the contribution is only perceived in unimodal functions, as can be seen in the results. For improving the performance in multimodal functions, we envision the addition of other knowledge sources, such as historical, topographic and domain knowledge, and fuzzy influence functions that make use of these knowledge sources to the CAEP framework as future work.

REFERENCES

Chung, C., Reynolds, R. G., 1997. *Fuzzy Approaches to Acquiring Experimental Knowledge in Cultural Algorithms*. In Proceedings of the 9th International Conference on Tools with Artificial Intelligence (ICTAI), IEEE Computer Society Washington, DC.

Fogel, D. B., 1995. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*, IEEE Press, Piscataway, NJ.

Fogel, D. B., Ghozeil, A., 1996. *Using Fitness Distributions to Design More Efficient Evolutionary Computations*. In, Proceedings of IEEE International Conference on Evolutionary Computation.

Reynolds, R. G., 1994. *An Introduction to Cultural Algorithms*. In Proceedings of the Third Annual Conference on Evolutionary Programming, February 24-26, San Diego, California.

Reynolds, R. G., Peng, B., Whallon, R., 2005. *Emergent Social Structures in Cultural Algorithms.* In Proceedings of the Annual Conference of the North American Association for Computational Social and Organizational Science, Notre Dame, Indiana.

Rodrigues, N. M., 2007. *Um Algoritmo Cultural para Problemas de Despacho de Energia Elétrica*, Master Dissertation, Universidade Estadual de Maringá, Maringá, Paraná, Brazil.

Saleem, S. M., Reynolds, R. G., 2000. *Cultural Algorithms in Dynamic Environments*. In Proceedings of the 2000 Congress on Evolutionary Computation, La Jolla, CA.

Saleem, S. M., Reynolds, R. G., 2001. *Knowledge-Based Solution to Dynamic Optimization Problems using Cultural Algorithms*. Ph.D. Thesis, Wayne State University, Detroit, Michigan.

Suganthan, P. N., Hansen, N., Liang, J. J., Deb, K., Chen, Y. P., Auger, A., Tiwari, S., 2005. *Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization,* Technical report, Nanyang Technological University, Singapore.

Zhu, S., Reynolds, R. G., 1998. *Fuzzy Cultural Algorithms with Evolutionary Programming for Real-Valued Function Optimization*, Ph.D. Thesis, Wayne State University, Detroit, MI.

DISTRIBUTED ARRIVAL TIME CONTROL FOR VEHICLE ROUTING PROBLEMS WITH TIME WINDOWS

Seok Gi Lee and Vittal Prabhu
Harold and Inge Marcus Department of Industrial and Manufacturing Engineering
The Pennsylvania State University, University Park, PA 16802, U.S.A.
sul201@psu.edu, prabhu@engr.psu.edu

Keywords: Distributed Arrival Time Control (DATC), Distributed Control, Scheduling, Dispatching.

Abstract: Competitiveness of a supply chains depends significantly on distribution center operations because it determines responsiveness and timeliness of deliveries to customers. This paper proposes a control algorithm for routing multiple out-bound trucks to customers spread over a wide geographical area, each occupying different volume in a truck, and having a different delivery time-window. Overall operations are also constrained by geographical locations of the customers in various zones and dissimilar truck capacities. Performance of the algorithm is tested using data from a distribution center located in Latin America.

1 INTRODUCTION

Transportation is the most expensive logistics activity. The overall goal in transportation should be to connect sourcing locations with customers at the lowest possible transportation cost within the constraints of the customer service policy (Edward H. Frazelle, 2002). Most important factors which can affect transportation cost are the number of trucks to deliver customer products, shipments allocation on trucks which decide a route of a truck and shipments loading on trucks as shown Figure 1. The truck (vehicle) routing problem has been recognized for over 40 years and is one of the most important factors in distribution and logistics. In particular, the importance of on-time delivery for customers is growing up according to various and complicated customer needs. In these conditions, finding solution optimally is very hard because of dynamics of a transportation environment. Many heuristics approach which can be broadly classified into two main classes, classical heuristics and meta-heuristics, have been proposed for vehicle routing problem in a last half century (Gilbert Laporte, Michel Gendreau, Jean-Yves Potvin, Frederic Semet, 2000).

Several meta-heuristic methods have been proposed to solve the vehicle routing problem. The important issues of meta-heuristics for the vehicle routing problems is how they can diversify search space and intensify routing solution to reduce transportation cost. For example, tabu search and simulated annealing algorithm tried to jump out of the local minimum by search its neighborhood space. To improve limitation of their neighborhood search space, some advanced methods were developed and plugged in search logic to diversify neighborhood search space (Haibing Li, Andrew Lim, 2003, J-F Cordeau, G Laporte and A Mercier, 2001).

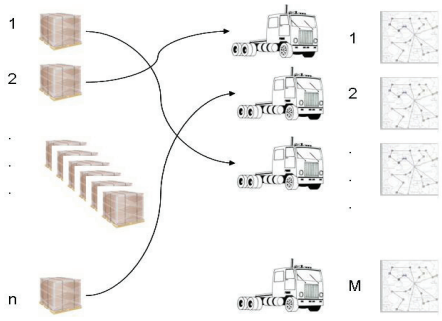


Figure 1: Transportation problem.

Furthermore, research for system dynamism has been conducted recently. Some measurements were proposed to explain how dynamic vehicle routing system (A Larsen, O Madsen and M Solomon, 2002, Larsen A, 2000). These measurements can play a crucial role in determining proper models or algorithms to solve vehicle routing problem according to the dynamic characteristics of the system.

In this paper, truck assignment and routing algorithm (TARA) is proposed to meet not only transportation cost needs but also customer service needs. Core of TARA algorithm is constructed based on distributed arrival time control (DATC) which is a feedback control-based scheduling approach that attempts to minimize the average of the square of the due-date deviation for Just-in-Time system (Hong, J., Prabhu, V. V., 2003).

2 PROBLEM DEFINETION

The problem, in terms of the distribution center and customer delivery, could be described as follow. The distribution center run 24 hours to process customers' transportation orders, but trucks operate loading and control jobs from 4 AM to 10 PM at the distribution center. During this time, there is no out-of-order of trucks which is used for delivery. All trucks are assumed to return to the distribution center and to be ready for loading at 4 AM.

The problem has three major constraints related with truck loading, delivery time and order processing. For truck loading constraints, each truck has own pallet capacity limit and maximum number of the customer order in a truck is four. Also, there are exclusive shipping requests which cannot share trucks with other shipments. For delivery time constraints, there are three different types of time window in this problem. In a sense, a time window implies the open time of a distribution center. A customer can request three type of time window. At first, for specific time of a specific date, delivery should be as punctual as possible. This is similar to Just-in-Time strategy with minimizing earliness and tardiness. Secondly, delivery could be done within a time window (w^{min}, w^{max}). Lastly, there could be no time requirement from customers. It means that a time window is within (0, 24) hour. Travel time among each customer's location including distribution center is shown in Table 1. For order processing constraints, customer orders arereceived every day, except on Sunday, until 6 PM. The scheduler generates the shipments of the next day. In other words, every customer order must be shipped the day after its arrival. Hence, if the distribution center cannot ship an order due to no available trucks, the customer order will be shipped the next earliest truck-available date, which will turn out to be a large deviation from the requested time window.

The truck assignment and routing algorithm is evaluated by two objective functions which are related with the trucking cost and customer service. The first objective function for trucking cost can be determined as follow:

$$\text{Minimize } \sum_{j=1}^{m} n_j \cdot C_j + \sum_{j=1}^{m} n_j \cdot F_j \qquad (1)$$

where $\forall j \in S$, S is the index set of trucks which are used for delivery, n_j is the number of trucks type j, F_j is the fixed cost for operating one truck and C_j is the trucking cost of truck type j. The second objective function related with customer service can be represented by time window violation cost and formulated as follow:

$$\text{Minimize } \sum_{i=1}^{n} ETC_i \qquad (2)$$

where $ETC_i = \alpha \cdot \max\{0, w_i^{min} - c_i\} + \beta \cdot \max\{0, c_i - w_i^{max}\}$ and c_i is completion time of customer order i. In this equation, α is the penalty cost for earliness and β represents the penalty cost for tardiness of time window for customer i.

Table 1: Customer location and travel time.

Location Code	L10001	L10002	L10003	•
L10001	0.00	0.42	0.44	•
L10002	0.42	0.00	0.50	•
L10003	0.44	0.50	0.00	•
L10004	2.05	2.16	1.83	•
L10005	0.75	0.80	0.49	•
L10006	0.84	0.92	0.59	•
•	•	•	•	•

3 DISTRIBUTED TIME CONTROL FOR TRUCK ASSIGNMENT

DATC is a closed-loop distributed control algorithm for manufacturing shop floor in which each part controller uses only its local information to minimize deviation from its part's due-date (Hong, J., Prabhu, V. V., 2003). In DATC, the integral control law is represented as follow:

$$a_i(t) = k_i \int_0^t (d_i - c_i(\tau))d\tau + a_i(0) \qquad (3)$$

where k_i is the controller gain, $a_i(0)$ is the arbitrary initial arrival time, d_i is the due-date and $c_i(\tau)$ is the predicted completion time for the ith job in the

system. In vehicle routing problem with time window, the controller gain value is defined by function of relationship between completion time and time window and it can be decided according to two cases, earliness and tardiness. When earliness occurs, the next arrival time of a job moves toward minimum value of time window by above integral control law. Similarly, in case of tardiness, the next arrival time moves toward maximum value of time window as shown in Figure 2.

Figure 2: Controller gain adaptation.

The momentum of each arrival time is controlled by the controller gain, k_i which is calculated differently according to the earliness and tardiness as described in equation (4) and (5).

$$k_i = KE_i = \frac{w_i^{\max} - c_i}{w_i^{\min} - c_i} \cdot k \,,\; c_i \le w_i^{\min} \qquad (4)$$

$$k_i = KT_i = \frac{c_i - w_i^{\min}}{c_i - w_i^{\max}} \cdot k \,,\; c_i \ge w_i^{\max} \qquad (5)$$

As a result, in case of earliness, equation (3) can be converted by equation (4) as follow:

$$a_i(t) = a_i(t-1) + KE_i \cdot \Delta \cdot \{w_i^{\min} - c_i(t-1)\} \qquad (6)$$

Similarly, equation (3) is changed by equation (5) for the tardiness case as follow:

$$a_i(t) = a_i(t-1) + KT_i \cdot \Delta \cdot \{w_i^{\max} - c_i(t-1)\} \qquad (7)$$

where $a_i(t)$ is the arrival time at tth time step, Δ is the time step and $c_i(t\text{-}1)$ is the completion time at (t-1)th time step.

Overall TARA procedure is described as follow:

STEP 1 : Initialize customer and truck parameters
a_i = minimum value of time window – t_{1i}
c_i, $u_j = 0$, $p_j = P_j$
$l_j = 1$ (Location 1 implies DC)
STEP 2 : Sort customer based on FCFS rule
STEP 3 : Truck assignment for each customer
For i = 1 to n Do
For j = 1 to m Do
j = Arg(min(E_j)), subject to $v_i \le p_j$
$e_j = E_j$, $p_j = p_j - v_i$, $l_j = l_i$
$u_j = e_j + v_i/R$, where R is the unloading rate in pallets/hour
$c_i = u_j$
$a_i = a_i + k_i \cdot \Delta \cdot (d_i - c_i)$
$E_j = u_j + t_{ji}$
STEP 4 : Compute summation of total earliness and tardiness for each i
STEP 5 : Initialize customer and truck parameters
STEP 6 : Go to STEP 2:

In these TARA steps, a_i is arrival time of customer i, c_i is delivery completion time of customer i, p_j is the number of pallets it can be loaded based on the current load of truck j, p_j is the maximum number of pallets it can be loaded by truck j, l_j is the current location of the truck j, l_i is the location of customer i, u_j is the last unloading time of truck j, v_i is the number of pallet of customer i and E_j is time consumption of truck j from current location to customer i. Also, by equation (6) and (7), k_i is KE_i in case of earliness or KT_i when the completion time is greater than maximum value of time window.

4 TRUCK ASSIGNEMNT AND ROUTING ALGORITHM PERFORMANCE

4.1 Performance Comparison

To measure the TARA performance, we used the following scalar equation (8) for mean squared due date deviation (MSD) which is used to characterize the global dynamics (Prabhu, V. V., 2003).

$$\text{MSD} = \sqrt{\frac{\sum_{i=1}^{n}(c_i - d_i)^2}{n}} \qquad (8)$$

Four customer demand sets which have 24, 31, 43 and 54 orders in them were used for test. These customer order data were real-world data used by one of global health and hygiene companies. The number of trucks was fixed as 12 and they have equal capacity to load 60 pallets. According to experiments, in case of set 1, 2 and 4, MSD became zero within 20th iteration. For set 3, time violation became the minimum value at 9th iteration. The MSD results for four kinds of data sets are described in Figure 3.

Furthermore, by using these experimental data, TARA performance was compared to dispatch rules, such as earliest due date (EDD), shortest processing

time (SPT) and latest processing time (LPT) as shown in Table 2. For the EDD rule, customer orders are arranged in ascending order by amount of time difference between each order's maximum time window and distance from the distribution center. Then each order is loaded in trucks one after another and finally, total MSD is calculated. For the SPT rule, orders are arranged in ascending order by distance between the distribution center and each order. Then, similar with the EDD rule, each order is loaded in trucks and total MSD is calculated. In case of the LPT rule, customer orders are arranged in descending order by same measurement with the SPT rule. The number of trucks was fixed as 12.

As a result, for average MSD of four data sets, TARA obtained 196% better result than the EDD rule. For the SPT and LPT rules, TARA showed approximately 199.6% improved results.

Experimental Results of minimum travel distance for each experiment set are shown in Table 3. Travel distance is estimated by assuming that the average trucking distance is 60 miles per hour. For set 1 and set 2 which have relatively small amount of customer orders, TARA with static controller gain have same or better performance than dynamic controller gain. In case of large amount of customer orders, TARA with dynamic controller gain gives relatively better performance. However, for almost all cases, dispatch rules have relatively better performance than TARA. This is because, basically, TARA controller proposed in this paper is designed to minimize tardiness and earliness of customer orders and it does not contain any device to consider travel distance.

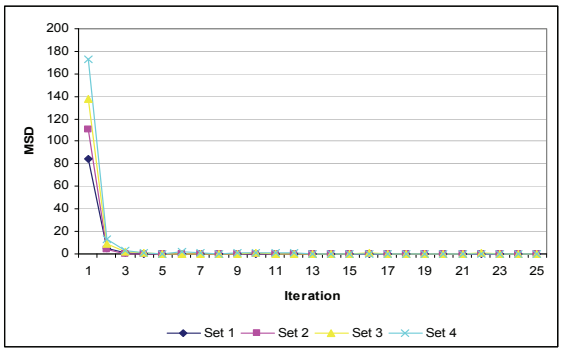


Figure 3: Mean squared due date deviation.

Table 2: Performance comparison – MSD.

Num of Order	TARA		EDD	SPT	LPT
	Static	Dynamic			
24	0.394	0.000	0.676	5.720	5.367
31	0.000	0.000	0.601	5.636	4.912
43	0.015	0.086	0.544	5.377	4.552
54	0.225	0.000	0.637	5.368	4.153

Table 3: Performance comparison – Travel Distance.

Num of Order	TARA		EDD	SPT	LPT
	Static	Dynamic			
24	1504	1504	1606	1409	1560
31	1635	1749	1861	1591	1905
43	2454	2427	2387	2105	2699
54	2959	2953	3320	2553	2926

4.2 Trucking Cost

Average total tardiness and average trucking cost were measured by varying the number of truck for TARA. To calculate trucking cost, fuel efficiency was estimated by 15 miles per gallon and fuel price was assumed by $3.5 per gallon. Also, fixed cost for a truck which has 1-ton capacity was assumed $0.25 per mile based on the Excel software program developed by the Texas A&M university (Ron Torrell, Willie Riggs and Duane Griffith). The number of customer orders and the controller gain value was set to 43 and 0.6. Experimental results for these two measurements are described in Figure 4. As the number of truck increases, average total tardiness increases. On the contrary, average trucking cost decreases as the number of truck increases.

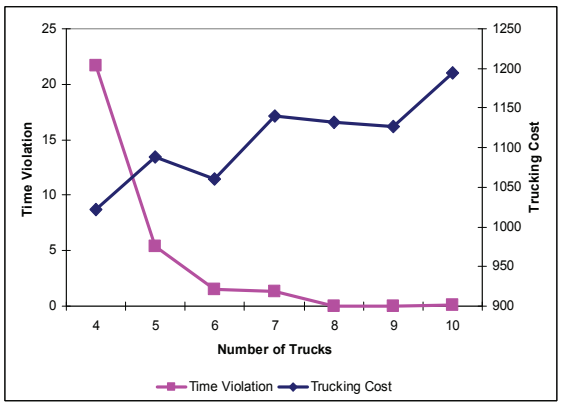


Figure 4: Trucking cost & time violation.

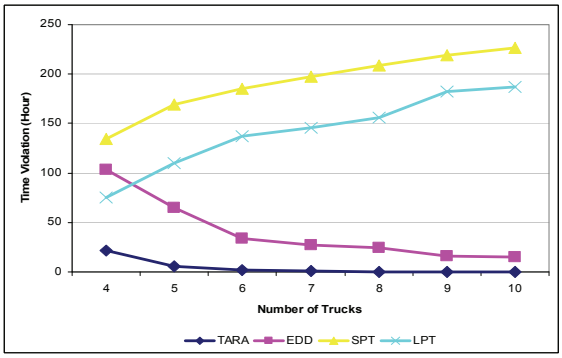


Figure 5: Time violation comparison.

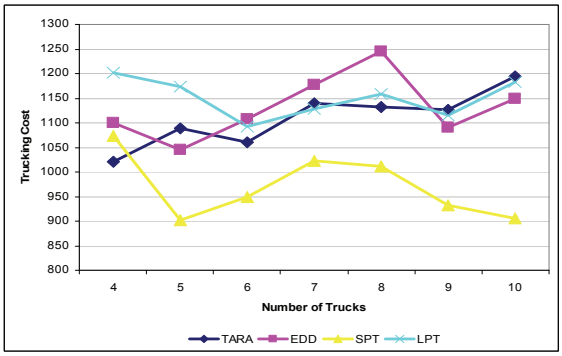


Figure 6: Trucking cost comparison.

Also, time violation and the trucking cost comparison among TARA and dispatch rules are shown in Figure 5 and Figure 6. In case of time violation, TARA and EDD had smaller time violation value as increasing the number of trucks. SPT and LPT, however, had increasing time violation value as increasing the number of trucks. In case of the trucking cost, it is hard to capture the relationship between the number of trucks and the trucking cost as shown Figure 6.

4.2 Dynamic Controller Gain Effects

As we explained in equation (4) and (5), TARA used dynamic controller gain which is changed by status of completion time. Actually, the basic distributed arrival time control updates the arrival time continuously with the static controller gain through fixed iteration. However, by changing the controller gain dynamically according to the status of completion time, earliness and tardiness, the convergence velocity and quality of MSD were improved as shown in Figure 7-10. In case of the 24 orders set (set 1), MSD from the dynamic controller gain reached zero at 14th iteration, but MSD from the static controller gain was greater than zero and it was not converged in zero. Similar results could be observed in other data sets. For the 31 (set 2), 54 (set 4) order sets, MSD from the dynamic controller gain were converged in zero at 8th and 21th iteration. In case of the 31 order set, however, MSD from the static controller gain was converged at 20th iteration and the other one was greater than zero and not converged. Although MSD by the dynamic controller gain of the 43 order set (set 3) was not converged, it was less than the result of the static controller gain and reached at minimum value faster.

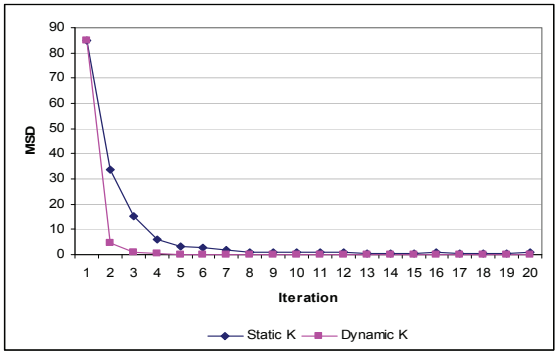


Figure 7: Dynamic controller gain effect – set 1.

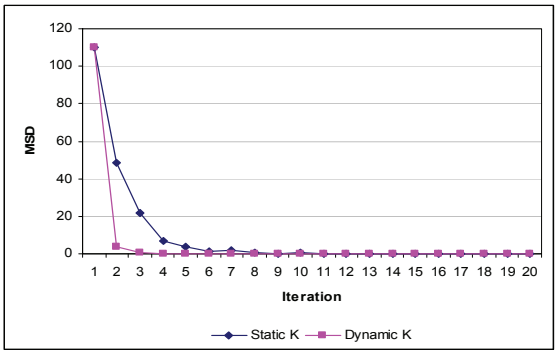


Figure 8: Dynamic controller gain effect – set 2.

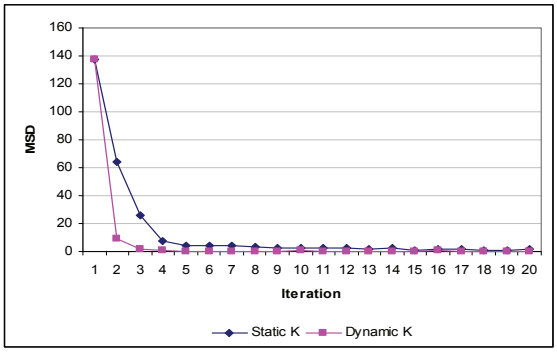


Figure 9: Dynamic controller gain effect – set 3.

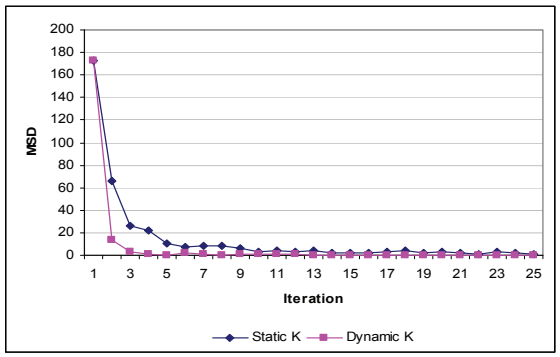


Figure 10: Dynamic controller gain effect – set 4.

5 CONCLUSIONS

Truck assignment and routing algorithm is an effective algorithm based on distributed arrival time control to solve the vehicle routing problem which has various delivery time windows of customers. In this work, TARA using the dynamic controller gain has been developed to determine the best vehicle routing plan for maximizing customer service level. Basically, the controller gain used in basic DATC is maintained static values through the whole algorithm processes. The dynamic controller gain, however, is updated continuously through whole iteration according to the result of the completion time, earliness and tardiness. Thus, we can improve not only the convergence velocity of the solution but also the quality of the solution compared with dispatch rules simultaneously.

ACKNOWLEDGEMENTS

The research was partially supported by the Marcus PSU/Technion Exchange Partnership.

REFERENCES

Prabhu, V. V., 2003, Stable Fault Adaptation in Distributed Control of Heterarchical Manufacturing Job Shops, *IEEE Transaction on Robotics and Automation*, 19(1), 142-149.

Hong, J., Prabhu, V. V., 2003, Modelling and performance of distributed algorithm for scheduling dissimilar machines with set-up, *International Journal of Production Research*, 41:18, 4357-4382.

Edward H. Frazelle, 2002, Supply Chain Strategy, McGraw-Hill.

Gilbert Laporte, Michel Gendreau, Jean-Yves Potvin, Frederic Semet, 2000, Classical and Modern Heuristics for the Vehicle Routing Problem, *International Transactions in Operational Research*, Res. 7, 285-300.

Haibing Li, Andrew Lim, 2003, Local Search with annealing-like restarts to solve the VRPTW, *European Journal of Operational Research*, 150, 115–127.

J-F Cordeau, G Laporte and A Mercier, 2001, A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows, *Journal of the Operational Research Society*, 52, 928-936.

A Larsen, O Madsen and M Solomon, 2002, Partially Dynamic Vehicle Routing Models and Algorithms, *Journal of the Operational Research Society*, 53, 637-646.

Larsen A, 2000, The dynamic vehicle routing problem, *PhD thesis*, Technical University of Denmark.

Ron Torrell, Willie Riggs and Duane Griffith, 2008, Find out how much truck costs per mile, *www.westernfamerstockman.com*.

COLLABORATIVE EXPLORATION IN GRID DOMAINS
Constructive Conjecture of a Polynomial Time Complexity

Yaniv Altshuler
Deutsche Telekom Laboratories, Ben Gurion University, Beer Sheva, Israel
yanival@cs.technion.ac.il

Alfred M. Bruckstein
Israel Institute of Technology, Computer Science Department, Technion, Technion City, Haifa 32000, Israel
freddy@cs.technion.ac.il

Israel A. Wagner
IBM Haifa Research Labs, Haifa, Israel
wagner@il.ibm.com

Keywords: Swarm algorithm, Decentralized search, Cooperative exploration, Grid exploration, Time complexity.

Abstract: This work discusses the problem of exploration of an unknown environment using a collaborative group of simple agents. While this problem was known to be of a non-polynomial time complexity, it was speculated in the past that in grid domains the completion time of this problem is much lower (although analytic proofs were not available hitherto). In this work we present a preliminary result concerning a constructive analytic constraint for guaranteeing that the time complexity of this problem in grid domains is indeed polynomial.

1 INTRODUCTION

In recent years significant research efforts have been invested in design and simulation of multi-agent robotics and intelligent swarms systems — see e.g. (Hettiarachchi and Spears, 2005; Wagner and Bruckstein, 2001; Steels, 1990) or (Arkin, 1990; Mataric, 1992; Haynes and Sen, 1986) for biology inspired designs (behavior based control models, flocking and dispersing models and predator-prey approaches, respectively), (Gerkey and Mataric, 2002; Rabideau et al., 1999; Thayer et al., 2000) for economics applications and (Chevallier and Payandeh, 2000) for a physics inspired approach).

Tasks that have been of particular interest to researchers in recent years include synergetic mission planning (Alami et al., 1998), swarm control (Mataric, 1994), human design of mission plans (MacKenzie et al., 1997), role assignment (Candea et al., 2001), multi-robot path planning (Yamashita et al., 2000), formation generation (Gordon et al., 2003), formation keeping (Balch and Arkin, 1998), exploration and mapping (Rekleitis et al., 2003), cleaning (Wagner et al., 2008) target tracking (Shucker and Bennett, 2005) and many more.

Unfortunately, the mathematical \ geometrical theory of such multi-agents systems is far from being satisfactory, as pointed out in (Beni and Wang, 1991; E.Bonabeau et al., 1999; Efraim and D.Peleg, 2007; Olfati-Saber, 2006) and many other papers.

One of the most interesting challenges for a robotics swarm system is the design and analysis of a multi-robotics system for searching and exploration (in either known or unknown areas). For example, works discussing cooperative searching tasks for static or dynamic targets can be found in (Altshuler et al., 2005b; Kerr and Spears, 2005; Passino et al., 2002; Polycarpou et al., 2001; Stone, 1975; Koopman, ; Vincent and Rubin, 2004) whereas examples for cooperative coverage of given regions are presented in (Rekleitisy et al., 2004; Rekleitis et al., 2005; Kong et al., 2006).

In this work we discuss the general problem of collaborative search of an unknown grid domains by a decentralized group of ant-like robots, while trying to establish an analytic proof for its time complexity. The strategy that will be used will based on the presentation of an “archetype problem”, to which an analytic upper bound over its completion time will be presented. From this bound, the time complexity of

the problem will be derived, and shown to be polynomial (when the bound predicts that the problem can be solved). Cases where the bound does not predict a successful completion of the mission are not treated here, although it is our belief that further analysis of the bound may yield a similar result for these cases as well.

Existing approaches and results are discussed in Section 2, while a conjecture concerning the time complexity in grid domains is presented in Section 3.

2 RELATED WORK AND MOTIVATION

In general, most of the techniques used for the task of a distributed coverage use some sort of cellular decomposition. For example, in (Rekleitisy et al., 2004) the area to be covered is divided between the agents based on their relative locations. In (Butler et al., 2001) a different decomposition method is being used, which is analytically shown to guarantee a complete coverage of the area. Another interesting work is presented in (Acar et al., 2001), discussing two methods for cooperative coverage (one probabilistic and the other based on an exact cellular decomposition). All of the works mentioned above, however, rely on the assumption that the cellular decomposition of the area is possible. This in turn, requires the use of memory resources, used for storing the dynamic map generated, the boundaries of the cells, etc'. As the initial size and geometric features of the area are generally not assumed to be known in advance, agents equipped with merely a constant amount of memory will most likely not be able to use such algorithms.

Surprisingly, while existing works concerning distributed (and decentralized) coverage often present analytic proofs for the ability of the system to guarantee the completion of the task (for example, in (Acar et al., 2001; Butler et al., 2001; Batalin and Sukhatme, 2002)), unfortunately, most of them lack analytic bounds for the coverage time (although in many cases an extensive amount of empirical results of this nature is made available). It can well be said that the time complexity of this problem, as well as the similar results for a variety of its derivatives, are still unavailable.

An interesting work to mention in this scope is this of (Svennebring and Koenig, 2004; Koenig and Liu, 2001), where a swarm of ant-like robots is used for repeatedly covering an unknown area, using a real time search method called *node counting*. By using this method, the robots are shown to be able to efficiently perform such a coverage mission, and analytic bounds for the coverage time are discussed. Based on a more general result for undirected domains shown in (Koenig et al., 2001), the following bound is given :

> The cover time of teams of ant robots (of a given size) that use node counting on strongly connected undirected graphs can be exponential in the square root of the number of vertices.

Namely :

$$f(k) = O(2^{\sqrt{S_0}}) \quad (1)$$

denoting the covering time of k robots by $f(k)$, and the initial area of the region to be explored by S_0.

It should be mentioned though, that in (Svennebring and Koenig, 2004) the authors clearly state that it is their belief that the coverage time of robots using nodes counting in grids is much smaller. This estimation is also demonstrated experimentally. However, no analytic evidence for this was available thus far. As grid domains are often used as an approximation for a problem in $\mathbf{R}^2$, the importance of such result is also likely to supersede its relevance to "purely discrete" problems.

3 TIME COMPLEXITY IN GRID DOMAINS

For dealing with the generic problem of exploring an unknown grid domain, we propose to discuss the "Dynamic Cooperative Cleaners" problem, a problem that requires several simple agents to clean a connected region of "dirty" pixels in $\mathbf{Z}^2$. A number of simple agents move in this dirty region, unaware of its size of shape, each having the ability to "clean" the place it is located in. Their goal being to jointly clean the given dirty region. The dynamic generalization of the problem involves a deterministic expansion of dirt in the environment, simulating spreading of *contamination*, or *fire*. By controlling the initial "contaminated" region any kind of search space in $\mathbf{Z}^2$ can be simulated (and any search space in $\mathbf{R}^2$ can be approximated). By employing the dynamic property of the contamination, other known problems can be derived. For example, collaborative hunting for evading targets (Altshuler et al., 2005b; Vincent and Rubin, 2004) may be simulated by using spreading contamination as the union of all possible target's locations.

It is also important to note that as the agents used in this problem are assumed to be as limited as possible (see more details in Section 3.1, any result obtained for this model will automatically be applicable for almost any other model of agents conceivable.

3.1 Cooperative Cleaners Problem

We shall assume that the time is discrete. Let the undirected graph $G(V,E)$ denote a two dimensional integer grid $\mathbf{Z}^2$, whose vertices (or "*tiles*") have a binary property called '*contamination*'. Let $cont_t(v)$ state the contamination state of the tile v at time t, taking either the value "*on*" or "*off*". Let F_t be the contaminated sub-graph of G at time t, i.e. : $F_t = \{v \in G \mid cont_t(v) = on\}$. We assume that F_0 is a single connected component. Our algorithm will preserve this property along its evolution.

Let a group of k agents that can move on the grid G (moving from a tile to its neighbor in one time step) be placed at time t_0 on F_0, at point $p_0 \in F_t$. Each agent is equipped with a sensor capable of telling the contamination status of all tiles in the digital sphere of diameter 7, which surrounds the agent. An agent is also aware of other agents which are located in these tiles, and all the agents agree on a common direction. Each tile may contain any number of agents simultaneously. Each agent is equipped with a memory of size $O(\log k)$ bits. When an agent moves to a tile v, it has the possibility of cleaning this tile (i.e. causing $cont(v)$ to become *off*. The agents do not have any prior knowledge of the shape or size of the sub-graph F_0 except that it is a single and simply connected component.

The contaminated region F_t is assumed to be coated at its boundary by a rubber-like elastic barrier, dynamically reshaping itself to fit the evolution of the contaminated region over time. This barrier is intended to guarantee the preservation of the simple connectivity of F_t, crucial for the operation of the agents, due to their limited memory. When an agent cleans a contaminated tile, the barrier retreats, in order to fit the void previously occupied by the cleaned tile. Every d time steps the contamination spreads. That is, if $t = nd$ for some positive integer n, then :

$$\forall v \in F_t \; \forall u \in 4-Neighbors(v) \; , \; cont_{t+1}(u) = on$$

While the contamination spreads, the elastic barrier stretches while preserving the simple-connectivity of the region, as demonstrated in Figure 1. For the agents who travel along the tiles of F, the barrier signals the boundary of the contaminated region.

The agents' goal is to clean G by eliminating the contamination entirely, meaning that : $(\exists t_{success} \; s.t \; F_{t_{success}} = \emptyset)$. In addition, it is desired that time $t_{success}$ will be minimal.

It is important to note that no central control is allowed, and that the system is fully decentralized (i.e. all agents are identical and no explicit communication between the agents is allowed). An important advantage of this approach, in addition to the simplicity of the agents, is fault-tolerance — even if almost all the agents cease to work before completion, the remaining ones will eventually complete the mission, if possible.

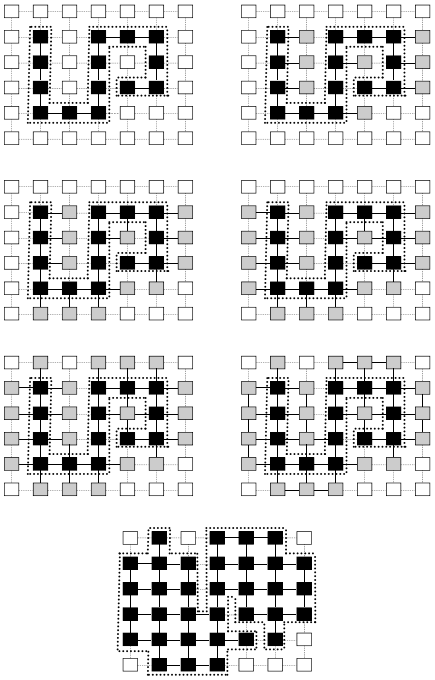


Figure 1: A demonstration of the barrier expansion process as a result of a contamination spread.

3.2 Existing Results

The cooperative cleaners problem was previously studied in (Wagner et al., 2008) (static version) and (Altshuler et al., 2005a; Altshuler et al., 2009) and others (dynamic version). A cleaning algorithm was proposed (used by a decentralized group of simple mobile agents, for exploring and cleaning an unknown "contaminated" sub-grid F, expanding every d time steps) and its performance analyzed. Following are two existing results containing upper bounds on the cleaning algorithm's completion time.

For some values of d and k, the cleaning might be completed in t_{static} time steps, even before the contamination is able to spread even once:

Theorem 1. *If $t_{static} < d$, then the region is known to be cleaned in less than d time steps (namely, before any contamination spread), where :*

$$t_{static} \triangleq \frac{8(|\partial F_0| - 1) \cdot (W(F_0) + k)}{k} + 2k$$

where $W(F)$ is the region's depth (the shortest path from some internal point in F to its surface,

for the internal point whose said shortest path is the longest) and ∂F denotes the boundary of F, defined as :

$$\partial F = \{v \mid v \in F \wedge 8-Neighbors(v) \cap (G \setminus F) \neq \emptyset\}$$

For a contaminated region F_0, for which the contamination is predicted to spread before being cleaned, then the following result is available :

Theorem 2. *Let* $\mu_{success} \triangleq \min\{x \in \{\mu_1, \mu_2\} | x > 0\}$ *where* μ_1 *and* μ_2 *are defined as :*

$$\frac{(A_4 - A_1A_3) \pm \sqrt{(A_1A_3 - A_4)^2 - 4A_3(A_2 - A_1 - A_1A_4)}}{2A_3}$$

where :

$$A_1 = \frac{c_0 + 2 - \gamma_2}{4}\ ,\ A_2 = \frac{c_0 + 2 + \gamma_2}{4}\ ,\ A_3 = \frac{8 \cdot \gamma_2}{d \cdot k}\ ,$$

$$A_4 = \gamma_1 - \frac{\gamma_2 \cdot \gamma}{d}\ \ ,\ \ \gamma_1 = \psi(1 + A_2) - \psi(1 + A_1)$$

$$\gamma_2 = \sqrt{(c_0 + 2)^2 - 8S_0 + 8)}$$

$$\gamma = \frac{8(k + W(F_0))}{k} - \frac{d - 2k}{|\partial F_0| - 1}$$

If such $\mu_{SUCCESS}$ *exists then* $t_{success} = (\mu_{success} - 1) \cdot d$.

Where c_0 is the circumference of the initial region F_0, and where $\psi()$ is the *Digamma* function (studied in (Abramowitz and Stegun, 1964)) — the logarithmic derivative of the Gamma *function*, defined as :

$$\psi(x) = \frac{d}{dx} \ln \Gamma(x) = \frac{\Gamma'(x)}{\Gamma(x)}$$

or as :

$$\psi(x) = \int_0^\infty \left(\frac{e^{-t}}{t} - \frac{e^{-xt}}{1 - e^{-t}} \right) dt$$

3.3 Grid Exploration Complexity

Using Theorem1, as $|\partial F_0| = O(S_0)$ and as $W(F_0) = O(\sqrt{S_0})$ we see that :

$$t_{static}(k) = O\left(\frac{1}{k} S_0^{1.5} + S_0 + k \right)$$

As for practical reasons we assume that $k < S_0$ we can see that :

$$t_{static}(k) = O\left(\frac{1}{k} S_0^{1.5} + S_0 \right)$$

and when the number of robots is independent in the size of the region, we can write :

Corollary 1. *If*

$$\frac{8(|\partial F_0| - 1) \cdot (W(F_0) + k)}{k} + 2k < d$$

Then

$$t_{static}(k) = O\left(S_0^{1.5} \right)$$

The analysis becomes slightly more complicated when $\frac{8(|\partial F_0| - 1) \cdot (W(F_0) + k)}{k} + 2k \geq d$, namely — when the contamination spreads before the robots had completed its cleaning. In this case, the cleaning time of the robots might be significantly longer, as clean tiles might become contaminated once again, while the robots are busy cleaning the evolving contaminated region. In order to analyze the complexity of the robots' cleaning time we shall use Theorem 2 :

Therefore :

$$t_{SUCCESS} =$$

$$d \cdot O\left(A_1 + \frac{A_4}{A_3} + \sqrt{A_1^2 + \frac{A_1A_4 + A_1 + A_2}{A_3} + \frac{A_4^2}{A_3^2}} \right) =$$

$$d \cdot O\left(A_1 + \frac{A_4}{A_3} + A_1 + \frac{\sqrt{A_3}\sqrt{A_1A_4 + A_1 + A_2}}{A_3} + \frac{A_4}{A_3} \right) =$$

$$d \cdot O\left(A_1 + \frac{A_4}{A_3} + \frac{\sqrt{A_1A_4} + \sqrt{A_1} + \sqrt{A_2}}{\sqrt{A_3}} \right) =$$

$$d \cdot O\left(c_0 + \gamma_2 + dk\frac{\gamma_1}{\gamma_2} + k\gamma + \sqrt{k}\sqrt{c_0 + \gamma_2}\sqrt{\gamma} \right) +$$

$$+ d \cdot O\left(\sqrt{k} \frac{\sqrt{c_0 + \gamma_2}\left(\sqrt{d\gamma_1} + 1\right)}{\sqrt{\gamma_2}} \right)$$

Taking into consideration that $(\gamma_2 \in \mathbb{R}) \Rightarrow (\gamma_2 \geq 1)$ and that $\gamma_2 = O(c_0)$ we can write the previous expression as :

$$d \cdot O\left(c_0 + dk\gamma_1 + k\gamma + \sqrt{k}\sqrt{c_0}\left(\sqrt{d\gamma_1} + 1 + \sqrt{\gamma} \right) \right) \tag{2}$$

As $|\partial F_0| = O(S_0)$ and as $W(F_0) = O(\sqrt{S_0})$ we can see that :

$$\gamma = O\left(\frac{\sqrt{S_0}}{k} + \frac{d + k}{\sqrt{S_0}} \right)$$

and when we require that the number of robots is independent of the size of the region, we can assume that $k < \sqrt{S_0}$ and write :

$$\gamma = O\left(\sqrt{S_0} + \frac{d}{\sqrt{S_0}} \right)$$

And as the delay between two spreads can assumed to be smaller than the size of the region, we can see that $\gamma = O\left(\sqrt{S_0}\right)$. In addition, remembering that $O(\sqrt{S_0}) \leq c_0 \leq O(S_0)$ we can rewrite the expression of equation 2 as follows :

$$t_{SUCCESS} = d \cdot O\left(c_0 + dk\gamma_1 + k\sqrt{S_0} + \sqrt{kdc_0\gamma_1} + \sqrt{k}S_0 \right) \tag{3}$$

Let us observe γ_1 :

$$\gamma_1 \triangleq \psi\left(1 + \frac{c_0 + 2 + \gamma_2}{4} \right) - \psi\left(1 + \frac{c_0 + 2 - \gamma_2}{4} \right)$$

As $(\gamma_2 \in \mathbb{R}) \Rightarrow (\gamma_2 > 0)$ we are guaranteed that $\left(1+\frac{c_0+2+\gamma_2}{4}\right) > 1$. Note that $\psi(1) = -\hat{\gamma}$ where $\hat{\gamma}$ is the *Euler-Mascheroni* constant, defined as :

$$\hat{\gamma} = \lim_{n\to\infty}\left[\left(\sum_{k=1}^{n}\frac{1}{k}\right) - \log(n)\right] = \int_1^{\infty}\left(\frac{1}{\lfloor x \rfloor} - \frac{1}{x}\right)dx$$

which equals approximately 0.57721. In addition, $\psi(x)$ is monotonically increasing for every $x > 0$. As we also know that $\psi(x)$ is upper bounded by $O(\ln(x))$ for large values of x, we see that :

$$-0.58 < \psi\left(1+\frac{c_0+2+\gamma_2}{4}\right) < O(\ln(c_0)) \quad (4)$$

We now need to present a bound for $\psi\left(1+\frac{c_0+2-\gamma_2}{4}\right)$. For this, it is sufficient to show that $\left(1+\frac{c_0+2-\gamma_2}{4}\right) > 1$. It is easy to see that this holds when $c_0 \geq \gamma_2$:

$$c_0 \,?\geq \gamma_2 \triangleq \sqrt{(c_0+2)^2 - 8\cdot(S_0-1)} \implies$$

$$c_0^2 \,?\geq (c_0+2)^2 - 8\cdot(S_0-1) = c_0^2 + 4c_0 - 4 - 8S_0 \implies$$

$$2S_0 + 1 \quad ?\geq \quad c_0$$

which of course holds, as $c_0 \leq 2S_0 - 2$. Therefore $\left(1+\frac{c_0+2-\gamma_2}{4}\right) > 1$ which implies :

$$-0.58 < \psi\left(1+\frac{c_0+2-\gamma_2}{4}\right) < O(\ln(c_0)) \quad (5)$$

Combining equations 4 and 5 we see that :

$$-O(\ln(c_0)) < \gamma_1 < O(\ln(c_0)) \quad (6)$$

Using equation 6 in order to rewrite equation 3 produces :

Conjecture 1.
$t_{SUCCESS} =$

$$d\cdot O\left(c_0 + dk\ln(c_0) + k\sqrt{S_0} + \sqrt{dkc_0\ln(c_0)} + \sqrt{k}S_0\right)$$

Comparing this to the bound of equation 1, we see that when successful completion of the mission by the cleaning protocol proposed is guaranteed, its time complexity is polynomial, as $c_0, d, k < O(S_0)$.

4 CONCLUSIONS

In this work the decentralized exploration problem using a group of simple, ant-like agents was discussed. A conjecture suggesting that the time complexity of this problem in grid domain is polynomial was presented. This was done by introducing a constraint which when followed guarantees a polynomial time completion. The importance of this result, as well as several related work were also discussed.

REFERENCES

Abramowitz, M. and Stegun, I. (1964). *Handbook of Mathematical Functions*, page 55. Applied Mathematics Series. National Bureau of Standards.

Acar, E., Zhang, Y., Choset, H., Schervish, M., Costa, A., Melamud, R., Lean, D., and Gravelin, A. (2001). Path planning for robotic demining and development of a test platform. In *International Conference on Field and Service Robotics*, pages 161–168.

Alami, R., Fleury, S., Herrb, M., Ingrand, F., and Robert, F. (1998). Multi-robot cooperation in the martha project. *IEEE Robotics and Automation Magazine*, 5(1):36–47.

Altshuler, Y., Bruckstein, A., and Wagner, I. (2005a). Swarm robotics for a dynamic cleaning problem. In *IEEE Swarm Intelligence Symposium*, pages 209–216.

Altshuler, Y., Wagner, I., and Bruckstein, A. (2009). Swarm ant robotics for a dynamic cleaning problem — upper bounds. *The 4th International conference on Autonomous Robots and Agents (ICARA-2009)*, pages 227–232.

Altshuler, Y., Yanovski, V., Wagner, I., and Bruckstein, A. (2005b). The cooperative hunters - efficient cooperative search for smart targets using uav swarms. In *Second International Conference on Informatics in Control, Automation and Robotics (ICINCO), the First International Workshop on Multi-Agent Robotic Systems (MARS)*, pages 165–170.

Arkin, R. (1990). Integrating behavioral, perceptual, and world knowledge in reactive navigation. *Robotics and Autonomous Systems*, 6:105–122.

Balch, T. and Arkin, R. (1998). Behavior-based formation control for multi-robot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939.

Batalin, M. and Sukhatme, G. (2002). Spreading out: A local approach to multi-robot coverage. In *6th International Symposium on Distributed Autonomous Robotics Systems*.

Beni, G. and Wang, J. (1991). Theoretical problems for the realization of distributed robotic systems. In *IEEE Internal Conference on Robotics and Automation*, pages 1914–1920.

Butler, Z., Rizzi, A., and Hollis, R. (2001). Distributed coverage of rectilinear environments. In *Proceedings of the Workshop on the Algorithmic Foundations of Robotics*.

Candea, C., Hu, H., Iocchi, L., Nardi, D., and Piaggio, M. (2001). Coordinating in multi-agent robocup teams. *Robotics and Autonomous Systems*, 36(2–3):67–86.

Chevallier, D. and Payandeh, S. (2000). On kinematic geometry of multi-agent manipulating system based on the contact force information. In *The Sixth International Conference on Intelligent Autonomous Systems (IAS-6)*, pages 188–195.

E.Bonabeau, M.Dorigo, and G.Theraulaz (1999). *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, US.

Efraim, A. and D.Peleg (2007). Distributed algorithms for partitioning a swarm of autonomous mobile robots. *Structural Information and Communication Complexity, Lecture Notes in Computer Science*, 4474:180–194.

Gerkey, B. and Mataric, M. (2002). Sold! market methods for multi-robot control. *IEEE Transactions on Robotics and Automation, Special Issue on Multi-robot Systems*.

Gordon, N., Wagner, I., and Bruckstein, A. (2003). Discrete bee dance algorithms for pattern formation on a grid. In *IEEE International Conference on Intelligent Agent Technology (IAT03)*, pages 545–549.

Haynes, T. and Sen, S. (1986). *Evolving Behavioral Strategies in Predators and Prey*, volume 1042 of *Lecture Notes in Computer Science*, chapter Adaptation and Learning in Multi-Agent Systems, pages 113–126. Springer, Berlin.

Hettiarachchi, S. and Spears, W. (2005). Moving swarm formations through obstacle fields. In *International Conference on Artificial Intelligence*.

Kerr, W. and Spears, D. (2005). Robotic simulation of gases for a surveillance task. In *Intelligent Robots and Systems (IROS 2005)*, pages 2905–2910.

Koenig, S. and Liu, Y. (2001). Terrain coverage with ant robots: A simulation study. In *AGENTS'01*.

Koenig, S., Szymanski, B., and Liu, Y. (2001). Efficient and inefficient ant coverage methods. *Annals of Mathematics and Artificial Intelligence*, 31:41–76.

Kong, C., Peng, N., and Rekleitis, I. (2006). Distributed coverage with multi-robot system. In *IEEE International Conference on Robotics and Automation*.

Koopman, B. The theory of search ii, target detection. *Operations Research*.

MacKenzie, D., Arkin, R., and Cameron, J. (1997). Multiagent mission specification and execution. *Autonomous Robots*, 4(1):29–52.

Mataric, M. (1992). Designing emergent behaviors: From local interactions to collective intelligence. In J.Meyer, H.Roitblat, and S.Wilson, editors, *Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, pages 432–441. MIT Press.

Mataric, M. (1994). *Interaction and Intelligent Behavior*. PhD thesis, Massachusetts Institute of Technology.

Olfati-Saber, R. (2006). Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420.

Passino, K., Polycarpou, M., Jacques, D., Pachter, M., Liu, Y., Yang, Y., Flint, M., and Baum, M. (2002). *Cooperative Control for Autonomous Air Vehicles*, chapter Cooperative Control and Optimization. Kluwer Academic, Boston.

Polycarpou, M., Yang, Y., and Passino, K. (2001). A cooperative search framework for distributed agents. In *IEEE International Symposium on Intelligent Control*, pages 1–6.

Rabideau, G., Estlin, T., Chien, T., and Barrett, A. (1999). A comparison of coordinated planning methods for cooperating rovers. In *Proceedings of the American Institute of Aeronautics and Astronautics (AIAA) Space Technology Conference*.

Rekleitis, I., Dudek, G., and Milios, E. (2003). Experiments in free-space triangulation using cooperative localization. In *IEEE/RSJ/GI International Conference on Intelligent Robots and Systems (IROS)*.

Rekleitis, I., New, A., and Choset, H. (2005). Distributed coverage of unknown/unstructured environments by mobile sensor networks. In *The Third MRS workshop*.

Rekleitisy, I., Lee-Shuey, V., Newz, A. P., and Choset, H. (2004). Limited communication, multi-robot team based coverage. In *IEEE International Conference on Robotics and Automation*.

Shucker, B. and Bennett, J. (2005). Target tracking with distributed robotic macrosensors. In *Military Communications Conference (MILCOM 2005)*, volume 4, pages 2617–2623.

Steels, L. (1990). Cooperation between distributed agents through self-organization. In Y.DeMazeau and J.P.Muller, editors, *Decentralized A.I - Proc. first European Workshop on Modeling Autonomous Agents in Multi-Agents world*, pages 175–196. Elsevier.

Stone, L. (1975). *Theory of Optimal Search*. Academic Press, New York.

Svennebring, J. and Koenig, S. (2004). Building terrain-covering ant robots: A feasibility study. *Autonomous Robots*, 16(3):313–332.

Thayer, S., Dias, M., Digney, B., Stentz, A., Nabbe, B., and Hebert, M. (2000). Distributed robotic mapping of extreme environments. In *Proceedings of SPIE*, volume 4195 of *Mobile Robots XV and Telemanipulator and Telepresence Technologies VII*.

Vincent, P. and Rubin, I. (2004). A framework and analysis for cooperative search using uav swarms. In *ACM Simposium on applied computing*.

Wagner, I., Altshuler, Y., Yanovski, V., and Bruckstein, A. (2008). Cooperative cleaners: A study in ant robotics. *The International Journal of Robotics Research (IJRR)*, 27(1):127–151.

Wagner, I. and Bruckstein, A. (2001). From ants to a(ge)nts: A special issue on ant—robotics. *Annals of Mathematics and Artificial Intelligence, Special Issue on Ant Robotics*, 31(1–4):1–6.

Yamashita, A., Fukuchi, M., Ota, J., Arai, T., and Asama, H. (2000). Motion planning for cooperative transportation of a large object by multiple mobile robots in a 3d environment. In *In Proceedings of IEEE International Conference on Robotics and Automation*, pages 3144–3151.

APPLICATION OF REACTIVE AGENTS CONCEPTS IN AUTOMATIC BINDINGS OF LONWORKS NETWORKS DEVICES

Miguel dos Santos Alves Filho, Rafael de Aquino Cunha and Carlos Eduardo Cugnasca
Escola Politécnica, University of São Paulo, Av. Prof. Luciano Gualberto, travessa 3 nº 158, São Paulo, SP, Brazil
miguel.alves@poli.usp.br, rafael.aquino.cunha@poli.usp.br, carlos.cugnasca@poli.usp.br

Keywords: LonWorks®, Multi-agents, Artificial Intelligence, Distributed systems, PBX, Telephony, Control networks.

Abstract: This paper aims to propose and test a new method to implement dynamic bindings in LonWorks® technology, allowing a new Distributed System of Private Telephone Switching (DSPTS), also developed with LonWorks® technology, to make their telephone links. In order to do this, a method for developing embedded systems and the reactive agent view was applied for each different device in this new system, offering a unique, practical and innovative solution for both, LonWorks® and PBX systems. This view allowed the implementation of intelligent and autonomous devices, specially in their internal process, providing satisfactory and more efficient results based on the DSPTS requirements. This work is the kick-off and the basis for developing new functions for telephone systems and control networks.

1 INTRODUCTION

In recent years, the paradigm of control systems design has been changed, moving from the traditional centralized architecture and proprietary technology to distributed and open architectures (Hur, Kim, Park, 2005)(Pu, Moor, 1998).

This change in paradigm takes the concept of network control, and brings some benefits to automation systems, such as reducing and simplifying cabling, increasing reliability and providing more options of manufacturers and integrators for the user to chose from (Echelon Corporation, 2007) (Hur, Kim, Park, 2005).

Aiming to exploit these advantages, a research project is presented that proposes to develop a Private Automatic Branch Exchange (PABX) (now also called Distributed System of Private Telephone Switching - DSPTS), with distributed architecture and implemented with the control networking technology LonWorks®. In this architecture, all telephone extensions and devices for interfacing with the telephone line are implemented as devices (also called nodes) of control network that communicate with each other, with the audio signals (voice) digitized and transferred via messages.

The standard method for using such a technology and the available tools to work with it provide only a static way for the realization of logical connections between devices (called bindings). It means that these connections are defined during the setup project and then will remain unchanged (Echelon Corporation, 1999).

Considering that a PABX system must open and close communication channels every time, it means that the logical connections between nodes are not pre-defined, but defined according to the need to establish a telephone connection. Thus the use of this control networks technology could not be exerted in its original form.

The purpose is to propose and analyze a solution based on the concepts of multi-agent systems to develop devices of this distributed PABX, using the LonWorks® technology, with self-management in a previously configured environment, enabling the creation of dynamic telephone links for the DSPTS.

The DSPTS characteristics of a distributed application stimulated the use of the multi-agent system concepts in its implementation (Durfee; Rosenchein, 1994).

1.1 LonWorks® Technology

LonWorks® emerged in the 1990s, developed by Echelon. This is a technology for automation networks, involving not only a communication

protocol, but also all the necessary infrastructure for the system development such as management tools, product development tools and network configuration tools (Echelon Corporation, 1999).

Its protocol was developed based on the OSI reference model, from the link layer to the session layer. It is an open and standardized protocol, ANSI / EIA 709.1, also known as LonTalk (Echelon Corporation, 2002).

An emphasis point of this protocol is its messages structure, which allows direct messaging between devices, broadcast messages and multicast messages. Furthermore, it provides system management messages, allowing a greater control of devices and even loads and develops programs by the network itself (Echelon Corporation, 2002).

In the implementation of control networks various media types can be used, as well as twisted pair, power network, optic fiber and wireless communication, which makes the technology very flexible (Echelon Corporation, 1999) (Chermont, 2007).

The LonWorks® control network nodes may be treated as objects that communicate through interfaces, also called functional profiles, which have network variables of the device in question. Through these variables, information may be shared with other devices (Xie, Pu, Moore, 1998).

The communication between the nodes is made through links between their network variables, a process called binding. From these, logical connections are used that allow messages to be sent whenever there is a change in the network variables values (Echelon Corporation, 1999).

Besides the LonWorks® technology design, Echelon has developed a chip called NeuronChip that is a full implementation of the LonTalk and provides the basis for any manufacturer to develop its products.

The NeuronChip consists of three processors working in parallel, two of them for the treatment of the communication protocol and the last one for the application program execution (Motorola, 1997).

The use of NeuronChip allows developers to treat the network variables as program variables, aside from the whole network communication, developing the programs associated with the application (Motorola, 1997). A specific programming language was developed for that technology, called NeuronC, which follows the ANSI C and is completely based on events (Echelon Corporation, 1995a).

1.2 Reactive Agents

Agents are autonomous entities that work together to achieve the same goal, being able to interact and to organize efficiently (Liviu, Sean, 2004).

They can also be perceived as real or virtual automatons that have knowledge of the environment in which they are inserted, and are able to perceive changes in that environment. They have knowledge about other agents, being able to communicate, to learn, to infer, to form groups or even to reject. Finally, they are able to decide, through its own observations, goals, knowledge and interactions (Wooldridge, Jennings, 2002).

The most common way to classify these agents divides them into two groups, considering the deliberation capacity, the environment perception and communication complexity: one is called cognitive agent and the other, reactive agent.

The cognitive agents are characterized by having very complex functions and by having models that require high processing capacity (Wooldridge, Jennings, 2002).

There are different internal architectures that can be used for the development of such agents, being one of the most used architecture known as “Beliefs, Desires, Intentions” (BDI) (Rao, 1996) (Rao; Georgeff, 1991). Its use is considered in systems that require the ability to exchange complex information, to own and build complete models of the world where they belong, their own and other agents, thus acting spontaneously, creating an organization that serves a purpose common to all of them (Steels, 1990).

A good example of the use of cognitive agents is presented by Bigham (2003), which consists of an antennas chain for mobile phones, each one with a manageable coverage area, so that intersections between them can be done and undone.

Within the group of reactive agents, there are those who are able to understand (though in a fairly limited way) and to react by acting on the environment in which they are inserted, through a pre-defined logic and always with a final goal that was set in project phase.

A striking feature of this model type is its simple communication way, which often occurs indirectly, through the environment itself (Wooldridge; Jennings, 2002). One advantage of these agents is the ease of implementation, which may be based on devices with less processing capacity and great limitation of energy.

As examples of its use, the study of the behavior of insects and their development processes may be

mentioned, as shown in Liviu and Sean (2004), or control network devices with low processing capacity. Finally, this class of agent is used in systems in which intelligence is expected to arise from the society overall behavior and not from each individual (Steels, 1990) (Castelfranchi, 1998) (Boissier; Demazeau, 1994).

It is worth mentioning that the software agents are used in systems known as multi-agents systems or distributed problem resolutions, which have as one of their characteristics the lack of centralized control, which fall within one of the DSPTS requirements as well their simplicity, allowing their implementation in the NeuronChips (Durfee; Rosenchein, 1994).

1.3 The DSPTS

One of the most important systems for any enterprise is the voice communication system.

Currently, these systems tend to be implemented with equipment called PABX, which are nothing more than core private telephony, which enables internal communication between all employees who have access to branches, and communication between them and the outside world (external links).

One important feature of this equipment is its centralized architecture, in which all telephone extensions and external lines must be connected to a central device, as illustrated in Figure 1. Among the main limitations of this architecture there is the limit of the expansion in the extensions number and external lines, and the need for a large amount of cabling, which makes it not very flexible to install and to change the branches positions.

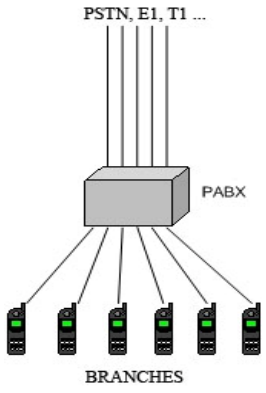


Figure 1: Typical PBX architecture.

The research presented in this article proposes a new architecture for the implementation of such systems, using the technology for control networks LonWorks®, allowing to remove the mentioned disadvantages of typical PABXs, as well the reduction and simplification of cabling, flexibility for changes, incremental growth (and associated investments), and greater system reliability (gaps in some nodes do not prevent the functioning of the rest of the system).

Moreover, this project aims to add greater intelligence to these systems, solving problems such as call direction, to decide which carrier should be used according to the call type, extensions calls prioritization to access the outside line, etc.

Figure 2 illustrates the architecture of the proposed system, in which the following DSPTS components are present:

- TLM – Trunk Line Module;
- E1M – E1 Module;
- ARM – Audible Response Module;
- DTM – Digital Telephony Module.

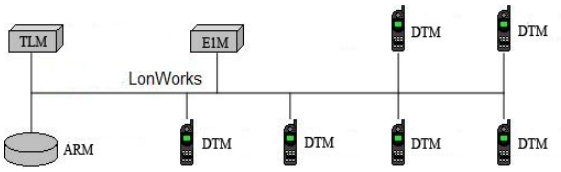


Figure 2: DSPTS distributed architecture.

Other features that motivated the choice of the LonWorks® technology was the possibility of the object oriented programming and the low jitter in message exchange, due to the robust features of the communication protocol.

For the voice message exchange, the project was specified for the use of voice compression algorithms, aimed at telephony. In this case, the G.729 ADPCM algorithm was used, which provides a good quality for telephony voice communication and uses a low bandwidth (16 kbps per communication channel).

Just as occurs in conventional PBX, it is necessary for the DSPTS to close communication channels, with the difference that in the first case the link is physical and in the second it is logical.

In the LonWorks® technology, this logical link is performed by means of bindings between the variables of two devices that, as reviewed above, are built with the tools in the market, during the system configuration. This makes these links static, preventing the intended application.

Thus, the possibility to perform bindings dynamically is fundamental for the DSPTS because that way two devices could exchange messages during a phone call, after which they would be free

to communicate with other devices and participate in other connections.

2 MATERIALS AND METHODS

The nodes developed have low processing capacity and low memory amount, and because it is a research project, in which new requirements can always be discovered during the evaluation of prototypes, an incremental development methodology has been adopted, but devised for smaller equipment.

Sequence diagrams were also used, quite common in program development and data flow diagrams, more used in projects with microcontrollers.

Figure 3 shows a representation of the equipment used in the system development, as well as the physical interconnection between them. In this assembly, an outside line Public Switched Telephone Network (PSTN) was used, connected to the TLM to make external calls.

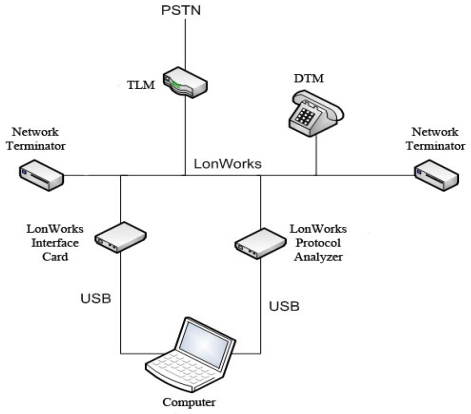


Figure 3: Experimental assembly.

NodeBuilder and LonMaker softwares were used for the program development. Both are provided by Echelon. LonTalk Protocol Analyzer (LPA) software, developed by Loytec, was also used (Loytec, 2008).

The TLM and DTM devices were developed with NeuronChip microcontroller, and meet the following project requirements:

- low overhead communication: to reduce the data traffic amount on the network, leaving a greater free bandwidth for the voice messages traffic.
- to use of the smallest number of entries in the tables used for device configuration: the device aims to keep compatible to be applied to other control network features.
- to use a telephony compatible time to the bindings configuration (or closing the phone link): avoid discomfort to the user, and maintain compatibility with already installed PABX systems.
- to dispense human operation and maintain the device full independence: ensures that the system does not require constant maintenance and keeps their integrity for long periods.

As a way to evaluate the solution proposed, the LonWorks® network analyzer was used to measure the total time consumed by the dynamic binding process proposed, comparing with the standard binding process, and checking whether the result meets the time requirements of the project, that is one second (to meet the standards of Brazilian telephony).

Several tests were also conducted, involving external calls request and calls receiving, forcing the process to be performed many times.

3 PROPOSED MODEL

As a way of meeting the specifications mentioned in the previous item, an extra software layer was added to each device, including the functionality of the device, and representing the reactive agent, as illustrated in Figure 4:

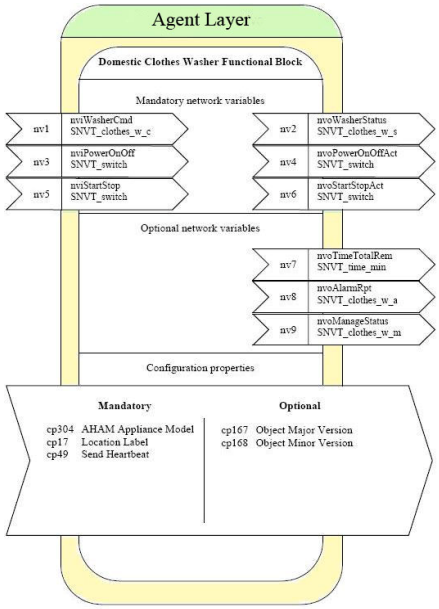


Figure 4: Reactive agent layer.

To facilitate the solution development, the problem was divided into three stages: the first one includes the handshake and the decision parameters to be used in the binding configuration, the second is to achieve the proper bindings, and the third stage involves the process for undoing the bindings, leaving the devices available for new connections.

3.1 Stage 1

To perform this stage, the device must be in an initial waiting state, in which the agent is active and attentive to any requests for connections from other agents.

When the device receives a call or request for assistance, the agents will exchange direct messages between themselves, adjusting the parameters needed to achieve the bindings.

This stage is completed when both have all the settings necessary for the binding to be achieved.

3.2 Stage 2

With the parameters adjusted, they should begin the configuration process. At this point, it is worth mentioning that each agent is responsible for conducting its own configuration process, not interfering with the process of another agent.

If there is any failure in the execution of this stage, the device sends a message of failure to the other agent, indicating that the process should be canceled.

After the setup process, they perform a check to ensure it was successful and that the communication can be initiated.

With a positive response of the final assessment, the exchange messages of telephony are initiated, including the whole call progress process and voice message exchange. At this time, the agent gets away from control, passing it to the basic operating software of the device.

3.3 Stage 3

When the call is finished, the agent takes back the control and performs all the configuration of the device, so that the bindings are undone and the device is ready to perform a new call.

At this stage, the communication between agents is not necessary, because it involves only the configuration of the internal tables of each device.

4 RESULTS ANALYSIS

The stage 1 obtained average time was approximately 23 ms. This time is due to the need of several message exchanges between the devices for the parameters to get adjusted.

Furthermore, depending on the case, it is necessary for the settings table to get swept several times, until the agents enter into a consensus about which values can be used.

The traditional method of conducting bindings, made from the LonMaker tool is naturally much faster, since it does not require the exchange of messages between devices and the configuration tool already knows the free values that can be used.

Various failures simulations were also performed, such as loss of communication during the process, and all tests showed that the solution is effective in failure recovering and left the devices ready for another attempt.

In stage 2, the average time needed to perform the required configurations was approximately 602 ms.

This time was measured from the receipt of the message containing the configuration values adjusted between the devices and the first call progress message.

This time is justified by the fact that the device must make all the configuration parameters in these tables that are stored in the EEPROM memory area, a memory type that demands a longer time for completing records.

When performing the same procedure with the LonMaker, the average time obtained was 1,983 ms. This long time is due to the fact that besides having to write data into EEPROM memory, all the configuration and verification is conducted through the control network, with device management messages.

Finally, the results analysis for stage 3 did not take into consideration the completion time, but if the devices were in a waiting state, as specified before.

After this step, one can verify that the bindings were completely undone and the devices were free to receive new connections. Figure 5 shows the devices tables at the end of stage 2.

The tables below, containing binding information between the device TLM (MLT) and DTM (MTD), show that the telephone link between them was made.

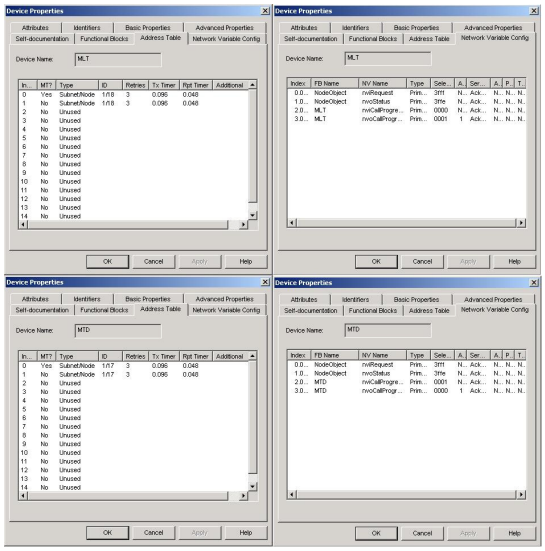


Figure 5: Table setup with bindings between devices.

Figure 6 shows the same table at the end of stage 3.

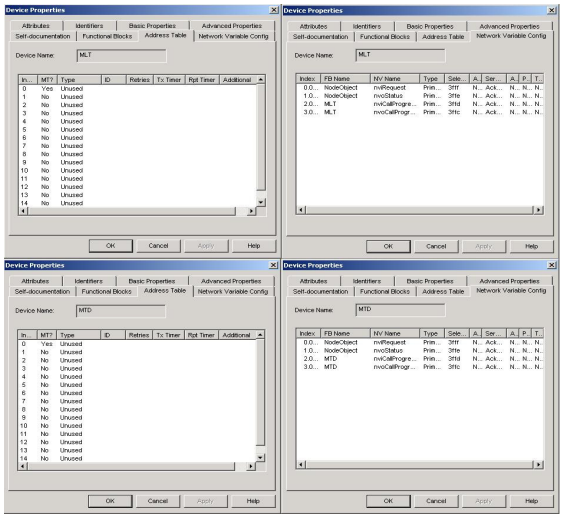


Figure 6: Configuration tables at the end of stage 3.

As seen, all the configuration entries have been deleted and, therefore, the devices are free for new connections.

5 CONCLUSIONS

The total mean obtained in completing the telephone link process, from the beginning of the stage 1 to the end of the stage 2, was approximately 625 ms, which is considered a good result for telephone systems. This result means that after the user enters a phone number or make a request for external line, he will hear the calling signal or external line tone signal after about 625 ms, time barely noticeable by the user.

At the end of stage 3, the devices had been effective in returning to the original state and ready to receive new calls.

From the point of view of multi-agents systems, a reactive agent was applied with reservations to the communication means used.

A characteristic of reactive agents is to have a primitive method of communication, often using only the very environment in which they are inserted (Wooldridge, Jennings, 2002). In this solution, the agents exchanged messages by a complex network protocol.

However, these agents did not have any model type: of the environment model, of other agents, and of their own model. They were not able to make inferences, to perform learning or to seek knowledge.

They only reacted to external stimuli as a previously implemented logic that could not be changed at runtime, which are features of a reactive agent (Wooldridge; Jennings, 2002).

Through the application agents' concepts, the presented solution meets the requirement of maintaining the devices autonomy, since they do not depend on external commands to configure themselves.

Also, no human intervention was required during the process execution, no form of communication used during the exchange of voice messages is LonWorks® standard and no overhead was added to the protocol. Only the minimum necessary parameters were used for the bindings configuration.

Thus, the conclusion is that all project requirements were met.

As an improvement to consider, there is the NeuronChip memory implementation. As LonWorks® technology considers that bindings are permanent, their settings are stored in an EEPROM area, which may support approximately 1,000,000 recordings (Motorola, 1997). For example, assuming that under heavy use 50 calls are made per day from a certain branch, which corresponds to 100 recording procedures in the EEPROM memory, the durability of the equipment would be approximately 27 years.

6 FUTURE WORK

In the context of the DSPTS project, one can imagine the possibility of applying cognitive agents

to solve greater complexity problems and that may require a system with greater integrity and decision capability.

As example of such problems, there is the decision of which attendant branch will take an incoming call for an installation where there are many attendants' branches. This could be solved through the use of auctions with pre-established metrics, method widely used in cognitive agents societies (Benisch et al., 2004).

For a broader case that involves the entire LonWorks® technology, the implementation and evaluation of new hardware for the network nodes can be proposed, enabling the storage of bindings information in RAM. This would remove the restrictions associated with the EEPROM memory, allowing faster bindings and a longer system life time.

A natural DSPTS continuity suggestion is related to the design of building automation systems based on LonWorks® technology that integrate the PABX functions.

ACKNOWLEDGEMENTS

The authors wish to thank the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), project number 05/56045-6, and the company Conceito Tecnologia, for the support granted to this research.

REFERENCES

Benisch, M., Greenwald, A., Grypari, i., Lederman, R., Naroditskiy, v., Tschantz, M., Botticelli. A Supply Chain Management Agent. Proc. 3rd. Int. Conf. On Autonomous Agents and Multi Agent Systems (AAMAS´04), ACM Press (New York), p. 1174-1181, 2004.

Bigham, J., Du, L. Cooperative Negotiation in a Multi-Agent System for Real-Time Load Balancing of a Mobile Cellular Network. In: Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems, 2., 2003, Melbourne (Australia). International Conference on Autonomous Agents. Nova York: ACM, 2003. p.568-575.

Boissier, O., Demazeau, Y. Asic: An Architecture for Social and Individual Control and its Application to Computer Vision. Modeling Autonomous Agents in a Multi-Agent World, v. 1069, 1994.

Castelfranchi, C. Modeling Social Action for AI Agents. Artificial Intelligence, Elsevier Science Publishers Ltd, v. 103, n. 1, p. 157-182, 1998.

Conceito Tecnologia. The company's website containing information on the LonWorks® technology and some examples of application. Available at: <http://www.conceitotecnologia.com.br>. Access at: 12 feb. de 2008.

Durfee, E. H., Rosenchein, J. S. Distributed Problem Solving and Multi-Agent Systems. Proceedings of the International Workshop on Distributed Artificial Intelligence, 1994.

Echelon Corporation. The company's website containing information on the LonWorks® technology. Available at: <http://www.echelon.com>. Access at: 17 de jun. de 2007.

Estados Unidos. Echelon Corporation. Neuron C programmer's guide. Palo Alto, 1995, 1 v.

Estados Unidos. Echelon Corporation. Introduction to the LonWorks® System, Palo Alto, 1999. 1 v. (078-0183-01A).

Estados Unidos. Echelon Corporation. LonMark Layer 1-6 Interoperability Guidelines, Palo Alto, 2002. 1 v.

Estados Unidos. Echelon Corporation. LonMaker User's Guide, Palo Alto, 2003. 3.1 v. (078-0168-02E). (a)

Estados Unidos. Echelon Corporation. NodeBuilder User's Guide. Palo Alto, Echelon, 2003. 1 v. (b)

Estados Unidos. Motorola. LonWorks® Technology Device Data, rev. 4, 1997.

Liviu, P., Sean, L. A Pheromone-Based Utility Model for Collaborative Foraging. Proc. 3rd. Int. Conf. On Autonomous Agents and Multi Agent Systems (AAMAS), p. 36-43, 2004.

Lonmark International. Site of standardized organ for LonWorks technology and products certification. Available at: <http://www.lonmark.org/>. Access at: 19 jan. 2008.

Loytec. The company's website. Available at: <http://www.loytec.com/english/products/lcore.htm>. Access at: 19 jan. 2009.

Rao, A. S. AgentSpeak(L): BDI Agents speak out in a logical computable language. Australian Artificial Intelligence Institute, 1996.

Rao, A. S., Georgeff, M. P. Modeling Rational Agents within a BDI-Architecture. Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91), USA, p. 473-484, 1991.

Shoham, Y. Agent-Oriented Programming. Journal of Artificial Intelligence, n. 60, p. 51-92, 1991.

Steels, L. Cooperation between distributed agents through self-organization. Intelligent Robots and Systems '90. 'Towards a New Frontier of Applications', Proceedings. IROS '90. IEEE International Workshop, Japan, p. 8-14, 1990.

Wooldridge, M., Jennings, N. Introduction to Multi-Agent Systems. USA: JASSS, 2002.

Xie, C., Pu, J.-S., Moore, P. R. A case study on the development of intelligent actuator components for distributed control systems using LONWORK neuron chips. Mechatronics v. 8, n. 2, p. 103-119, 1998.

FAULT DETECTION AND DIAGNOSIS IN A HEAT EXCHANGER

Juan C. Tudon Martinez, Ruben Morales-Menendez and Luis E. Garza Castañon
Tecnológico de Monterrey, Av. E. Garza Sada 2501, 64849, Monterrey N. L., México
{A00287756,rmm}@itesm.mx, legarza@itesm.mx

Keywords: Dynamic principal component analysis, Diagnostic observers, Fault detection, Fault diagnosis.

Abstract: A comparison between the Dynamic Principal Component Analysis (DPCA) method and a bank of Diagnostic Observers (DO) under the same experimental data from a shell and tube industrial heat exchanger is presented. The comparative analysis shows the performance of both methods when sensors and/or actuators fail. Different metrics are discussed (i.e. robustness, quick detection, isolability capacity, explanation facility, false alarm rates and multiple faults identifiability). DO showed quicker detection for sensor and actuator faults with lower false alarm rate. Also, DO can isolate multiple faults. DPCA required a minor training effort; however, it cannot identify two or more sequential faults.

1 INTRODUCTION

Early detection and diagnosis of abnormal events in industrial processes represent economic, social and environmental profits. Generally, the measuring and actuating elements of a control system fail causing abnormal events. Thus, when the process has a great quantity of sensors or actuators, the Fault Detection and Isolation (FDI) task is very difficult.

Most of the existing FDI approaches for Heat Exchangers (HE), are based on quantitative model-based methods. In (Ballé et al., 1997), fuzzy models are used to generate residuals; since each fault has an unique residual incidence, it is possible the fault isolation. Similarly, a residual generator is proposed to create fault signatures in (Krishnan and Pappa, 2005). Generalized Likelihood Ratio is frequently used to estimate the fault magnitude from a residual generation (Aitouche et al., 1998). On the other hand, a particle filtering approach for predicting the probability distribution of different heat exchanger states (faults) is proposed in (Morales-Menendez et al., 2003).

A comparative analysis between two FDI systems in an industrial HE is proposed. One of them is based on the Dynamic Principal Component Analysis (DPCA) and another one on a bank of Diagnostic Observers (DO).

Some researches are related to this work. Recently, DPCA and correspondence analysis (CA) have been compared (Detroja et al., 2005). CA shows a greater efficiency of fault detection in terms of the shorter detection delay and lower false alarm rates; however, CA needs a greater computational effort. An adaptive standardization of the DPCA has been proposed for MIMO systems (Mina and Verde, 2007); simulation results allow to detect faults and avoid normal variations in process signals.

An adaptive observer of a nonlinear discrete-time system with actuator faults is proposed in (Caccavale and Villani, 2004). Using process linear models, a dynamic observer detects malfunctions caused by measurement and modeling errors (Simmani and Patton, 2008). In order to detect multiple faults in a process, a set of unknown input-observers can be used, each one of them is sensitive to a fault while insensitive to the remaining faults (Verde, 2001).

The aforementioned works were implemented under different types of faults and processes; then, a comparison under same experimental data in an industrial HE is considered.

This paper is organized as follows: Section 2 formulates the DPCA approach. Section 3 describes the steps for designing a set of DO. Section 4 describes the experimental system. Section 5 discusses the results. Finally, conclusions of this work are presented.

2 DPCA

Let $\mathbb{X}$ be a matrix of m observations and n variables collected from a real process. This data set represents the normal operating conditions. $\overline{\mathbb{X}}$ is the scaled data matrix and $\bar{x}$ is a vector containing mean (μ)

of each variable. Such that $\bar{x} = (\frac{1}{m})\mathbb{X}^T\mathbf{1}$ and $\bar{\mathbb{X}} = (\mathbb{X} - \mathbf{1}\bar{x}^T)\mathbb{D}^{-1}$ where $\mathbb{D}$ is a diagonal matrix containing standard deviation (σ) of each variable and $\mathbf{1}$ is a vector of elements equal to 1.

When the system has a dynamic behavior, the data present a serial and cross-correlation among the variables. This violates the assumption of normality and statistical independence of the samples. To overcome these limitations, the column space of the data matrix $\mathbb{X}$ must be augmented with a few past observations for generating a static context of dynamic relations.

$$\bar{\mathbb{X}}_{\mathbb{D}} = \begin{array}{c}[X_1(t)X_1(t-1),\ldots,X_1(t-w),\ldots \\ X_n(t)X_n(t-1),\ldots,X_n(t-w)]\end{array} \quad (1)$$

where w represents the quantity of time delays. By performing PCA on the augmented data matrix, a multivariate auto regressive model is extracted directly from the data (Ku et al., 1995). For a multivariate system, the process variables can have different ranges of values, thus the data matrix $\mathbb{X}_{\mathbb{D}[m\times(n\times w)]}$ must be standardized. With the scaled data matrix, a set of a smaller number ($r < n$) of variables is searched through the process of decomposing the variance in the data. r must preserve most of the information given in these variances and covariances.

The dimensionality reduction is obtained by a set of orthogonal vectors, called *loading vectors* (p), which are obtained by solving an optimization problem involving maximization of the explained variance in the data matrix by each direction (j) with $t_j = \bar{\mathbb{X}}p_j$; the maximal variance of t_j must be computed from:

$$max(t_j^T t_j) = max(p_j^T \bar{\mathbb{X}}^T \bar{\mathbb{X}} p_j) = max(p_j^T \mathbb{A} p_j) \quad (2)$$

Such that $p_j^T p_j = 1$. Solving the optimization problem through the Singular Value Decomposition (SVD), the eigenvalues λ_j of the matrix $\mathbb{A}$ are computed from,

$$(\mathbb{A} - \lambda_j \mathbb{I})p_j = 0 \quad \text{for j=1,...,n} \quad (3)$$

where, $\mathbb{A}$ represents the correlation matrix of the data matrix $\bar{\mathbb{X}}$, and $\mathbb{I}$ is a $n \times n$ identity matrix. Using the new orthogonal coordinate system, the data matrix $\bar{\mathbb{X}}$ can be transformed into a new smaller data matrix $\mathbb{T}$, called *scores matrix*.

$$\mathbb{T}_{[m\times r]} = \bar{\mathbb{X}}_{[m\times n]}\mathbb{P}_{[n\times r]} \quad (4)$$

where, $\mathbb{P}$ represents the obtained loading vectors of the SVD with the most significant eigenvalues λ_j. As this transformation is a rotation matrix, it holds $\mathbb{P}^T\mathbb{P} = \mathbb{I}$. Therefore also $\bar{\mathbb{X}} = \mathbb{T}\mathbb{P}^T$ is valid. Thus, PCA decomposes the matrix $\bar{\mathbb{X}}$ as,

$$\bar{\mathbb{X}} = t_1 p_1^T + t_2 p_2^T + \ldots + t_r p_r^T \quad (5)$$

The matrix $\mathbb{T}$ can be back-transformed into the original data coordination system as,

$$\mathbb{X}^*_{[m\times n]} = \mathbb{T}_{[m\times r]}\mathbb{P}^T_{[r\times n]} \quad (6)$$

2.1 FDI using DPCA

The normal operating conditions can be characterized by T^2-statistic (Hotelling, 1993). Equation (7) allows to generate online the T^2-statistic based on the first r loading vectors (*principal components*).

$$T^2 = x^T_{[1\times n]}\mathbb{P}_{[n\times r]}\Lambda^{-1}_{[r\times r]}\mathbb{P}^T_{[r\times n]}x_{[n\times 1]} \quad (7)$$

where, x is a new measurement vector taken online and Λ is a diagonal matrix which contains first r eigenvalues of the correlation matrix ($\mathbb{A}$). If the value of T^2-statistic stays within its control limit then, the status of the process is considered normal (Ku et al., 1995). Thus, a fault occurs, when a value of T^2-statistic is greater than its control limit (T^2_α).

$$T^2_\alpha = \frac{(m-1)r}{(m-r)}F_\alpha(r, m-r) \quad (8)$$

where, $F_\alpha(r, m-r)$ is the F-distribution with r and $m-r$ degrees of freedom with $100\alpha\%$ of confidence.

Due T^2-statistic only detects variation in the direction of the first r principal components, Jackson *et al.* (Jackson and Mudholkar, 1979) propose to monitor the variation in the residual space (components associated with the smallest singular values) using Q-statistic for helping to fault detection. Both statistics must detect a fault, however they have not the same resolution in the deviation when the fault occurs. Similarly to T^2-statistic, when a value of Q-statistic is greater than its threshold (Q_α) indicates the occurrence of a fault. The values of Q-statistic and its control limit can be calculated through the equations:

$$Q = [(\mathbb{I} - \mathbb{P}\mathbb{P}^T)x]^T[(\mathbb{I} - \mathbb{P}\mathbb{P}^T)x] \quad (9)$$

$$Q_\alpha = \theta_1\left[\frac{h_0 c_\alpha\sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0(h_0-1)}{\theta_1^2}\right]^{\frac{1}{h_0}} \quad (10)$$

where, $\theta_i = \sum_{j=r+1}^{n}(\lambda_j)^{2i}$, $h_0 = 1 - \frac{2\theta_1\theta_3}{3\theta_2^2}$ and c_α is the normal deviation corresponding to $(1-\alpha)$ percentile.

Once a fault is detected, the next step is the isolation. In order to determine which variable is the most relevant to cause the fault, the use of contribution plots has been proposed (Miller et al., 1998). Contribution plots quantify the error of each process variable when the process is not in normal operating conditions. The variable which shows the highest contribution (Con_i) to the error is isolated and associated as the most relevant to the fault which has occurred.

$$Con_i = \frac{R_i^2}{\sum_{j=1}^{r} R_j^2} \quad (11)$$

where, R_i represents the residue in the residuals space (Isermann, 2006). The residue $\mathbb{R}$ can be calculated by

subtracting the back-transformation data (equation 6) to scaled data matrix ($\bar{\mathbb{X}}$),

$$\mathbb{R}_{[m\times n]} = \bar{\mathbb{X}}_{[m\times n]} - \mathbb{T}_{[m\times r]}\mathbb{P}^T_{[r\times n]} \tag{12}$$

where $\mathbb{P}$ contains the loading vectors corresponding to components with the smallest singular values.

3 DESIGN OF A BANK OF DO

As the state observer compute the error between the process states and adjustable model states, it can be used as a further alternative for model-based fault detection. The discrete state space model which can describe the process dynamic is,

$$\begin{aligned} x_p(k+1) &= \mathbb{G}x_p(k) + \mathbb{H}u(k) \\ y(k) &= \mathbb{C}x_p(k) \end{aligned} \tag{13}$$

A state observer for unmeasurable state variables can be represented as

$$\begin{aligned} \tilde{x}_o(k+1) &= \mathbb{G}\tilde{x}_o(k) + \mathbb{H}u(k) + \mathbb{K}_e[y(k) - \hat{y}(k)] \\ \hat{y} &= \mathbb{C}\tilde{x}_o(k) \end{aligned} \tag{14}$$

where, $\mathbb{K}_e$ is the observer feedback matrix.

3.1 FDI using a Bank of DO

The error of the observer can be computed as:

$$x_p(k+1) - \tilde{x}_o(k+1) = (\mathbb{G} - \mathbb{K}_e\mathbb{C})[x_p - \tilde{x}_o] \tag{15}$$

Defining $e(k) = x_p - \tilde{x}_o$ as the error vector, the predicted error can be calculated as

$$e(k+1) = (\mathbb{G} - \mathbb{K}_e\mathbb{C})e(k) \tag{16}$$

The dynamic behavior of the error $e(k)$ is determined by the eigenvalues of $\mathbb{G}$-$\mathbb{K}_e\mathbb{C}$. If the matrix $\mathbb{G}$-$\mathbb{K}_e\mathbb{C}$ is a stable matrix, the error vector will converge to zero for any initial error $e(0)$.

When an unknown input (fault) changes the process normal operation, the error signal called *residual*, should be different to zero. Therefore, if the residual is close to zero (i.e. noise with $\mu = 0$ and $\sigma = 1$), the process variable is into its normal operating condition, called nominal behavior.

If the process is affected by several faults, it is possible to use a bank of DO for identification of different faults. All DO are designed from different fault models and they are sensitive to any fault except the used fault for their design.

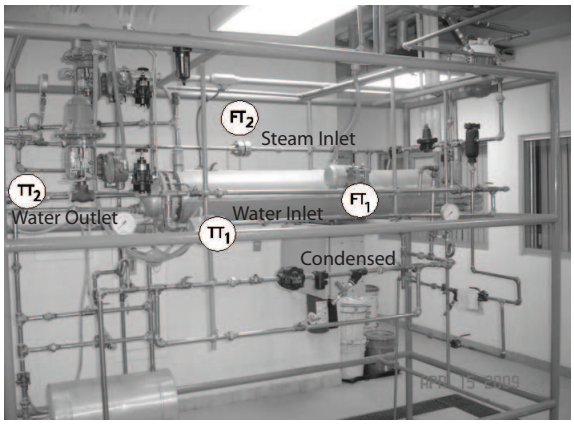


Figure 1: Experimental System.

4 EXPERIMENTAL SYSTEM

An industrial shell-tube heat exchanger is used, whose characteristics of non-linearity and slow transient response are the most relevant, see Figure 1.

Faults in sensors and actuators, called *soft faults*, have been implemented in additive form. Also, the process always was free of disturbances.

DPCA used 1 second as sample time delay; and 1900 measurement data of each sensor were taken.

$$x(t) = [FT_2(t)\ FT_1(t)\ TT_1(t)\ TT_2(t)] \tag{17}$$

where, FT_1 and FT_2 are flow transmitters and TT_1 and TT_2 are temperature transmitters.

In case of diagnostic observers, 5 seconds of sample time are used to obtain the state space models for each faulty condition. The observer feedback matrix in each observer is designed via pole placement with closed loop poles close to origin in the discrete space.

Four types of additive soft faults will be implemented: abrupt fault in sensors, gradual fault in sensors, abrupt faults in actuators and multiple faults in sensors, Table 1.

Table 1: Types of faults in the sensors.

Sensor	Abrupt fault	Gradual fault (slope)
FT_1	6% (5σ)	$0.1\%/sec$
FT_2	8% (5σ)	$0.1\%/sec$
TT_1	2°C (8σ)	$0.1°C/sec$
TT_2	2°C (8σ)	$0.1°C/sec$

Five types of faults were implemented in the steam and water control valves, Table 2.

Table 2: Types of faults in actuators.

Case	Status of the steam valve	Status of the water valve
0	normal (70%)	normal (38%)
1	low pressure (60%)	normal (38%)
2	high pressure (80%)	normal (38%)
3	normal (70%)	low pressure (28%)
4	normal (70%)	high pressure (48%)

5 RESULTS

5.1 DPCA Approach

Taking one sample time delay of each measurement, it is possible to explain a high quantity of variance including the possible auto and cross correlations. The normal operating conditions can be explained with 5 principal components (99.95%).

When an abrupt fault was implemented in the TT_2 sensor at time 105, the Figure 2(left plot) shows that Q and T^2 statistics clearly overshoot their control limits. Figure 2(right lot) shows how the contribution plot helps correctly with the fault isolation. The 78% of total error corresponds to outlet temperature signal.

Figure 3 (left plot) shows a gradual fault in the TT_2 sensor at time 200. Q and T^2 statistics overshoot their control limits and indicate the fault detection after 14 and 10 seconds respectively once the fault has occurred. Figure 3(right plot) shows that 64% of total error corresponds to outlet temperature signal.

For actuator faults, independently if the bias is positive or negative, there is a reaction in both statistics. When T^2 and Q statistics overshoot their control limits, the fault is detected (Figure 4).

Using contribution plots, for the cases 1 and 2, the steam flow signal has the greatest error contribution followed by the outlet temperature signal (Figure 5). This result is right because the faults are associated to changes in the pressure of steam valve (negative and positive respectively). Similarly, the water flow signal has the greatest contribution to the error when the water valve is affected by a pressure change.

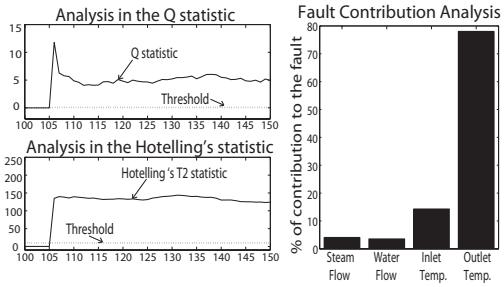


Figure 2: FDI analysis for an abrupt fault in the outlet temperature sensor using DPCA.

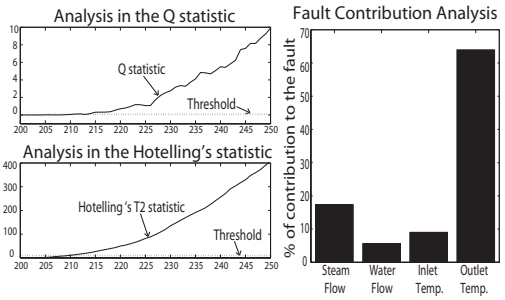


Figure 3: FDI analysis for a gradual fault in the outlet temperature sensor using DPCA.

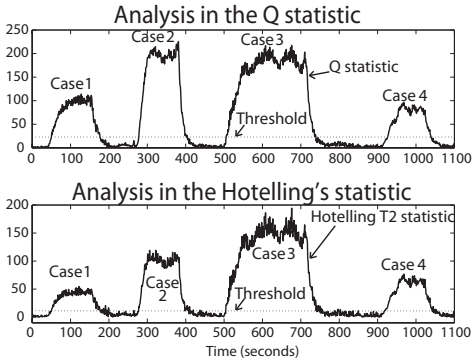


Figure 4: Fault detection for actuator faults using DPCA.

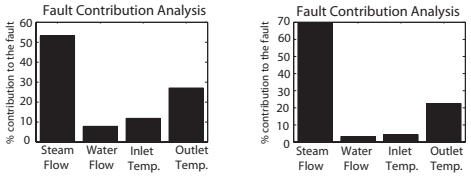


Figure 5: Results in actuators: case 1(left), case 2(right).

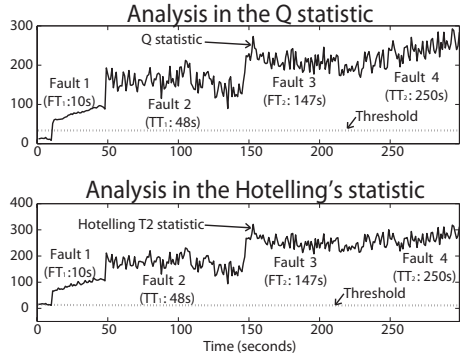


Figure 6: Fault detection using DPCA under multiple faults.

Finally, multiple faults have been activated sequentially at different times. Figure 6 shows the performance of DPCA; each fault presents its activation time. Both statistics overshoot their control limits when the fault 1 has occurred at time 10. When the

remainder of the faults were introduced, the statistics stay inside their control limits; however, they move more away from their thresholds. None of the statistics comes back to its normal status since none of the faults was deactivated. Figure 7 shows that it is not possible to isolate multiple faults since contribution plots can not associate the error to a specific variable.

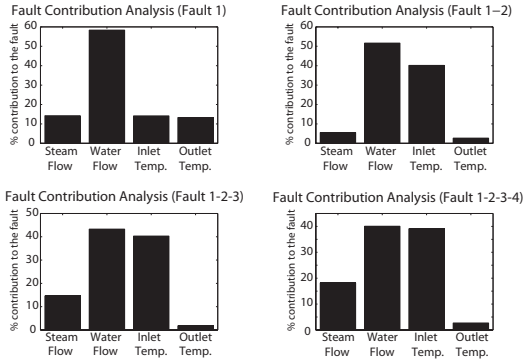


Figure 7: Diagnostic result for multiple faults in all sensors.

5.2 DO Approach

In order to distinguish different fault conditions, a bank of four DO was designed (i.e. water flow, steam flow, outlet temperature and inlet temperature).

When an abrupt fault is implemented in the TT_2 sensor, the outlet temperature residue is the unique signal which does not change its nominal behavior whereas the remainder of the residues are deviated negatively 1.5 units at time 10 when the fault is activated, Figure 8(top plot). Thus, it is possible to associate the fault to the TT_2 sensor. Same FDI result is obtained when a gradual fault is implemented in the TT_2 sensor. Figure 8(bottom plot) shows the fault detection after 5 seconds once the fault has occurred.

Figure 9 shows the performance of DO for faults in actuators Table 2. When is implemented a fault in the water control valve, independently of the bias direction, the water flow residue does not change its behavior from its nominal value; whereas, the remainder of the residues are deviated. Similarly, when is implemented a fault in the steam control valve, the steam flow residue does not change its behavior.

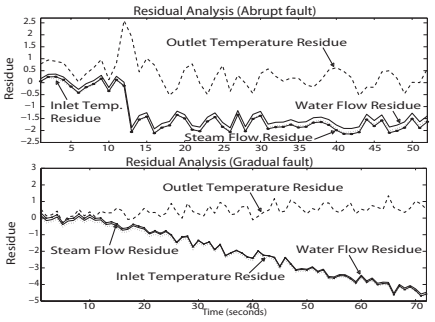


Figure 8: FDI analysis for an abrupt fault (top plot) and gradual fault (bottom plot) in the TT_2 sensor using DO.

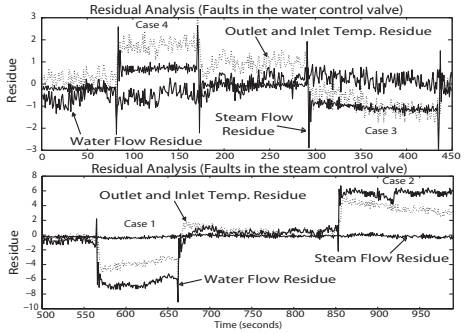


Figure 9: FDI analysis for actuator faults using the DO.

Figure 10 shows the FDD result using a set of DO when multiple faults have been activated sequentially at different time instants. It is important to note that only one signal is not deviated from its behavior when is introduced any abrupt sensor fault. The residual signal which does not change its behavior is associated to the occurred fault.

Comparison of the Methods. According to the Table 3, DO shows a quicker detection than DPCA when is implemented a gradual fault in a sensor signal. In this work, the gradual faults are added to a signal and only the deviations about the normal operating point are analyzed as residuals. In all fault cases, it is easy to explain the fault propagation using both FDI methods, i.e. the explanation facility metric is achieved.

Contribution plots indicate which variables are hypothetically more associated to the fault since it is possible that more fault cases are involved. On the other hand, a set of DO can correctly isolate a fault if all fault models and the model of the normal operating condition are known with high reliability. For faults in actuators, the normal operating conditions change in more than two sensors and the diagnosis task can

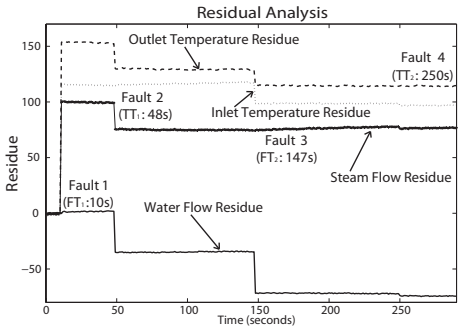


Figure 10: FDI result using DO under multiple faults.

Table 3: Comparison of DPCA and DO approaches.

	DPCA				DO			
Metrics	Abrupt Fault(TT_2)	Gradual Fault(TT_2)	Actuator faults	Multiple faults	Abrupt Fault(TT_2)	Gradual Fault(TT_2)	Actuator faults	Multiple faults
Detection (s)	0	$10-14$	$9-18$	0	0	5	$5-8$	0
Isolation	✓	✓	✓	-	✓	✓	✓	✓
Explanation	✓	✓	✓	-	✓	✓	✓	✓
False alarm (%)	0	0	14.13	0	0	0	9.94	0

be complicated. In this work, DO shows a quicker detection (i.e. almost the half of detection time) than DPCA when faults in both actuators are implemented at different times. For these faults, DO presents a lower false alarm rate than DPCA (Table 3).

On the other hand, both FDI methods can detect multiple faults which are implemented in all sensors. However, DPCA can not isolate correctly when several faults have been implemented. According to computational requirements, the design of DO needs greater computational resources. The training stage of this method is more complicated than the DPCA training; DO requires firstly a reliable ARX model which must be translated to a state space model. Furthermore, each fault case must be modeled in a particular state space model. Once the fault model is known with high reliability, is designed a state observer; particularly in this work all models (fault cases and normal operating) are obtained in parallel. On the other hand, the DPCA training is quickly executed once historical data of the normal operating point are known.

6 CONCLUSIONS

A comparison between the Dynamic Principal Component Analysis (DPCA) and a set of Diagnostic Observers (DO) under same experimental data from an industrial Heat Exchanger (HE) is presented. DPCA do very well on fast detection of abnormal situations, it is easier to implement in industrial applications. A process model was not required; however, a broad acquisition of the historical measurements is needed. Respect to false alarm rate, DPCA showed 42% more of false alarms than DO for actuator faults.

DO presents a quicker detection than DPCA ($[4-10]$ seconds lower), DO requires an accurate state space model of the process. Furthermore, each fault case must be modeled. If the model is not reliable, DO can not detect a fault correctly. Due to HE is inherently a nonlinear system, it is more difficult to implement a FDI method based on quantitative models. Finally, DPCA can not identify multiple faults whereas DO can.

REFERENCES

Aitouche, A., Maquin, D., and Busson, F. (1998). Multiple Sensor Fault Detection in Heat Exchanger Systems . In *Proc. of Int. Conf. on Ctrl. Appl.*, pages 741–745, Trieste, Italy.

Ballé, P., Fischer, M., Füssel, D., and Isermann, R. (1997). Integrated Control, Diagnosis and Reconfiguration of a Heat Exchanger . In *American Control Conference*, pages 922–926, Albuquerque, New Mexico.

Caccavale, F. and Villani, L. (2004). An Adaptive Observer for Fault Diagnosis Nonlinear Discrete-Time Systems. In *American Control Conference*, pages 2463–2468, Boston, Massachusetts.

Detroja, K., Gudi, R., and Patwardhan, S. (2005). Plant-wide Detection and Diagnosis using Correspondece Analysis. *Control Engineering Practice*.

Hotelling, H. (1993). Analysis of a Complex of Statistical Variables into Principal Components. *J. Educ. Psychol.*, 24.

Isermann, R. (2006). *Fault-Diagnosis Systems*. Springer, Germany, 1^{st} edition.

Jackson, J. and Mudholkar, G. (1979). Control Procedures for Residuals Associated with Principal Component Analysis. *Technometrics*, 21:341–349.

Krishnan, R. and Pappa, N. (2005). Real Time Fault Diagnosis for a Heat Exchanger A Model Based Approach. In *IEEE Indicon Conference*, pages 78–82, Chennai, India.

Ku, W., Storer, R., and Georgakis, C. (1995). Disturbance Detection and Isolation by Dinamic Principal Components Analysis. *Chemometrics and Intelligent. Lab. Syst.*, 30:179–196.

Miller, P., Swanson, R., and Heckler, C. (1998). Contribution Plots: A Missing Link in Multivariate Quality Control. *Appl. Math. and Comp. Sci.*, 4(8):775–792.

Mina, J. and Verde, C. (2007). Fault Detection for MIMO Systems Integrating Multivariate Statistical Analysis and Identification Methods . In *American Control Conference*, pages 3234–3239, New York City, USA.

Morales-Menendez, R., Freitas, N. D., and Poole, D. (2003). State Estimation and Control of Industrial Processes using Particles Filters. In *American Control Conference*, pages 579–584, Denver, Colorado, USA.

Simmani, S. and Patton, R. (2008). Fault Diagnosis of an Industrial Gas Turbine prototype Using a System Identification Approach. *Ctrl. Eng. Prac.*, (16):769–786.

Verde, C. (2001). Multi-leak Detection and Isolation in Fluid Pipelines . *Control Eng. Practice*, 9:673–682.

SEMANTIC SUPPORT FOR RESOURCE-CONSTRAINED ROBOT SWARM

Xiang Su, Jukka Riekki and Janne Haverinen
Intelligent Systems Group and Infotech Oulu, University of Oulu, FIN-90014, Finland
Xiang.Su@ee.oulu.fi, Jukka.Riekki@ee.oulu.fi, Janne.Haverinen@ee.oulu.fi

Keywords: Robot Swarm, Entity Notation, Lightweight Data Representation, Ontology, Inference.

Abstract: Semantic Web technology could offer lots of intelligent functionality to multi-robot systems. But limited processing power and storage capability of unsophisticated robots do not necessary allow them to support and process Semantic Web technology. In this paper, we propose a novel solution to provide semantic support for resource-constrained robots. Entity Notation is a lightweight data representation which can be employed to transfer data between resource-constrained robots and intelligent applications at server side. Resources-constrained robots only need to handle the lightweight Entity Notation while intelligent applications handle the more advanced knowledge representation. When the Entity Notation is used, the transfer between the robots and applications is unambiguous and lossless. In this way, an ontology and ontology-based inference at server side can improve the capabilities of the robot swarm. We present a simulator and discuss the future work.

1 INTRODUCTION

Research in multi-robot systems is producing more and more robot swarm systems containing up to hundreds of autonomous robots. Robot swarms can be utilized in numerous interesting domains, like space exploration, search and rescue operations, cleaning, and other everyday applications. Compared with individual complex autonomous robots, swarms of simple and cost-efficient robots, provide robustness against failure of individual robots and the power of parallel operation.

To build the complex and intelligent structure of a robot swarm, coordination among spontaneous swarms, like stigmergy (Holland and Melhuish 1999), or environment-supported coordination is essential. In this paper, we present how semantic support can be provided for resource-constrained robots to facilitate their coordination. Basically, the semantic support enables reasoning the actions for achieving the swarm's goal.

Semantic Web technology is an ideal candidate for implementing the reasoning, when information can be expressed using a Semantic Web-based formal context model – an ontology model (Studer, Benjamins and Fensel 1998). But traditionally, Semantic Web technology needs more processing power and memory than small robots have. Hence the robots are not capable to processing knowledge representations such as Resource Description Framework (RDF) (W3C.org 2004), not to mention running the advanced reasoners or other applications utilizing the representations.

This paper focuses on how resource-constrained robots can be connected to Semantic Web–based intelligent applications that have the required resources for processing the knowledge representations and running the reasoners. We consider how information can be transferred from resource-constrained robots to an advanced knowledge representation, how actions can be deduced, and how the actions can be transferred to the robots performing them.

For transferring information from resource-constrained robots to intelligent systems we propose a lightweight representation called Entity Notation (EN). The key idea is that the robots handle the lightweight EN while the intelligent systems handle the advanced knowledge representation – and an unambiguous lossless transform can be performed between these two representations. We develop for the knowledge representation a context information ontology for describing robots, sensors, persons, and other important entities of an indoor office environment. This ontology allows us to use Semantic Web technologies in deducing actions for the robots. We utilize rule-based inference for

deducing actions for robot swarm.

The rest of this article is organized as follows. Section 2 introduces the architecture of EN-based communication. Section 3 presents the Entity Notation in brief. Section 4 presents the context information ontology model. Section 5 describes the rules and our inference mechanism. Section 6 presents the simulator for testing the system. Section 7 introduces the related work and Section 8 concludes the paper and suggests future work.

2 ENTITY NOTATION-BASED COMMUNICATION ARCHITECTURE

Figure 1 presents a complete loop of EN-based communication architecture for a robot system. EN packets are utilized in all communication acts between a robot swarm and a swarm server. This robot swarm system consists of cleaning robots, RFID tags, sensors, an air conditioner, and the swarm server. A cleaning robot can measure the light level and the temperature of a room and read data from and write data to RFID tags. Robots and sensors can communicate with the swarm server. The swarm server includes ontology models, EN composer/decomposer, and inference engine components. It provides semantic support to robots by receiving data from robots and sending the actions to be performed back to corresponding robots. This EN-based feedback loop is a key feature of adaptive robot system.

In the application scenario shown in Figure 1, the RFID tag in one room is marked with the timestamp of cleaning work inside this room. When a cleaning robot goes inside this room, it reads the RFID tags of the room, and measures whether light is on in this room. This information can be transferred to the swarm server as EN packets. The swarm server decomposes the EN packets and transfers them to RDF statements in a lossless way. When the context information ontology model is supplied with new RDF statements, the reasoner is triggered, the deduced statements are composed into EN format and sent to the devices. Deduced statements can be some information to a specific device, like message "Cleaning robot521 cleans the room TS354" should be sent to Cleaning robot521. The statements can also be some general information, like "Room TS354 is clean." should be sent to all cleaning robots subscribed room cleanness information. In the process of compose/decompose EN packets, we use communicative acts ontology model to standardize the communicative acts between robots and swarm server, which could be attached as the first EN packet to indicate the type of the following packet sequence. For example, the 'ifRoomSituation' packet will indicate that the following packet will inform the situation of a specific room, while the 'ifCleanRoom' packet will follow with a packet to inform a specific cleaning robot an action of cleaning a room.

Another use case of EN-based communication in figure 2 is that, a temperature sensor measures the temperature in room TS354, and sends the measurement data to the same swarm server by utilizing EN. The reasoner could deduce that "Air conditioner of TS354 adjusts the temperature to 23 ºC." or "TS354 has normal temperature."

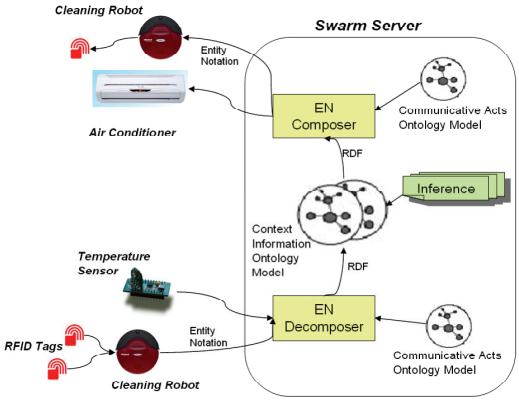


Figure 1: Loop of EN-based Communication.

3 ENTITY NOTATION

Entity Notation is introduced briefly in this section and more detailed description can be found in the publication by Riekki, Su and Haverinen (2008). EN was designed as a lightweight representation for exchanging information between nodes of a distributed system. We started to develop EN as a general representation that can be used by any resource-constrained device, but now we are focusing on mobile robots. It's not a protocol, hence we assume EN packets are payload of some protocols delivering the packets, like HTTP or TCP/IP.

The most important goals for developing EN are that, on the one hand, it can be used over simple communication links by resource-constrained devices; on the other hand, the packet content can be used in a straightforward fashion by intelligent system. For example, it should be possible to

transform the packets into RDF format in an unambiguous fashion.

We achieve these goals as follows. To support resource-constrained robots and limited communication links, EN offers short packets that in their simplest form can be less than 40 bytes long. To allow the packet content to be utilized by Semantic Web applications, the EN offers complete packets that can be transferred to advanced knowledge representations – RDF unambiguously and losslessly. Hence, packet content can be delivered to a reasonor processing knowledge in Web Ontology Language (OWL) format. Complete packets and short packets can be transferred between each other straightforwardly and unambiguously.

The complete EN packet format closely follows the RDF format, which contains a sequence of (subject, predicate, object) triplets. The EN, as its name implies, describes entities. An entity is some identifiable whole, physical or digital. For example, a robot, a sensor, an RFID tag, a sending node, a recipient, a person, a measurement, a message, and a user terminal are entities.

An EN packet is a sequence of entity descriptions. Each description specifies the type of an entity, a unique entity identifier, and a number of property triplets (name, type, value). XML Schema types are used for describing the data type of property value. An entity description is of the form:

```
[EntityTye EntityID
PropertyName PropertyType PropertyValue
…
PropertyName PropertyType PropertyValue
]
```

An entity description contains a set of triplets about a single entity. *EntityId* specifies the subject, *PropertyName* is a predicate, and *PropertyValue* is an object. *EntityType* and *PropertyType* specify the types of the corresponding subject and object. It is necessary that elements in EN packet are self-describing. *PropertyValue* in EN can be an identifier of the other entity, and additional statements can give information about those entities. This allows relations to be represented. The relations can be between physical entities, e.g. a robot is in a room. Or, the relations can be between digital entities. For example, the first entity of a message can specify a list and refer to all list member entities.

Here is an example in which a room entity description refers to a temperature sensor in that room. For convenience, in this and all the following examples we replace "http://ee.oulu.fi/o" by "EE" and "http://www.w3.org" by "W3":

```
 [EE#Room EE#room1879
 EE#hasTempSensor EE#EntityId
EE#tempSensor234
]

[EE#TempSensor EE#tempSensor234
 EE#tempValue W3/2001/XMLSchema#float
"23.5"
 EE#timestamp
W3/2001/XMLSchema#dateTime
"2007-10-22T12:00:00-17:00:00"
]
```

The type EntityID is an exception in the type system as all other types are XML Schema types. EntityId specifies the corresponding property value to be an entity identifier – not a literal URI (that would be the XML Schema type anyURI). Literals are enclosed in double quotes, as some data types such as base64Binary data type can contain white space but not double quotes. EN has the ability to carry any primitive data type defined in XML Schema types, including text data, binary data, et al.

A complete packet can be transferred to a RDF description in a straightforward fashion because of their natural relationship. The entity type can be represented as a resource class (using the rdf:type property) and the entity identifier is the identifier of the resource. Property names and values can be mapped directly, and the property type can be represented by adding a rdf:datatype attribute to the property element. Here is the RDF graph corresponding to the example above.

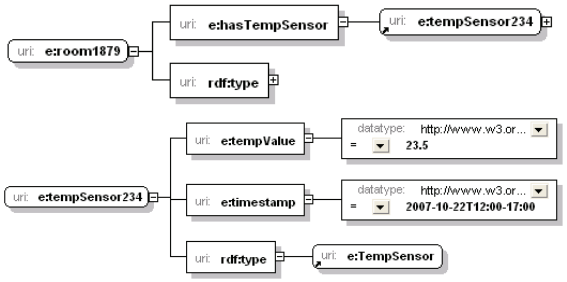


Figure 2: RDF graph representing a sensor's data package.

Complete EN packets can be quite long because of meaningful type names and URIs. To shorten the complete packets, we suggest templates and prefixes. The basic idea is that a template contains a description of the constant part of a packet and placeholders for the variable parts that differ in each packet. The packet sent over the communication link needs to contain only the template identifier and the changing information. A complete packet can then be assembled by replacing the template's

placeholders with the values contained in the package. Prefixes are used where the values are URIs. Such as the above example of room entity and temperature sensor inside, if we have for the packet the templates:

```
 [EE#Room EE#room1879
 EE#hasTempSensor EE#EntityId ?1
]
```

and

```
]
[EE#TempSensor EE#tempSensor234
 EE#tempValue W3/2001/XMLSchema#float
 ?1
 EE#timestamp
 W3/2001/XMLSchema#dateTime ?2
]
```

And then template has the identifier `urn:uuid:ad7g38` and `urn:uuid:5e76af`, we can represent the packets as:

```
[urn:uuid:ad7g38 EE#tempSensor234]
[urn:uuid:5e76af
    "23.5"
    "2007-11-22T12:00:00-17:00:00"]
```

Robots and swarm server can have predefined templates for each packet they can send. Usage of short packets can either be decided by the engineer at design time or the robots can use complete packets by default and agree the short format during communication. The negotiation of EN packets is not in the scope of this paper.

4 THE CONTEXT INFORMATION ONTOLOGY MODEL

Context information gathered from environment can increase a robot swarm's capabilities by enabling them to adapt to the conditions of environment. By context, we refer to any information that can be used to characterize the situation of an entity (Dey and Abowd 2000).

An ontology-based approach lets us describe contexts semantically, and share common understanding of context among devices, users, and services. Here, ontology refers to the formal, explicit description of concepts, which are often conceived as a set of entities, relations, instances, functions, and axioms. Our context information ontology model facilitates context reasoning, context sharing and reuse. For example, the model can be extended with information about other devices, or it can be merged with another ontology to describe a larger domain. We will introduce ontology-reasoning and general-purpose rule-based reasoning in the next section.

Figure 3: Main catalogue of Context Information Ontology Model.

In our scenario, we assume a smart house which is equipped with networked robots and devices, such as cleaning robots, RFID tags, temperature sensors, and air conditioners. As shown in Figure 3, context information ontology models the basic concepts for our scenario, like device, environment, person, time and location; describes properties and relationship between these concepts. The context information ontology can be extended to different environment easily depending on other application scenarios.

5 INFERENCE MECHANISM

As discussed above, inference engine will be triggered when the information from robots is added into context information ontology. New knowledge will be produced by reasoner and distributed to devices in EN format. If the EntityID of the deduced statement is one device, EN composer will add one header packet before other packets and deliver them to that device. If its EntityID is not device, the swarm server will distribute it to robots have subscribed this information.

Currently, our inference mechanism supports OWL Lite inference and rule-based inference. OWL Lite inference supports constructs of describing classes, properties, and also relationship between classes and properties. The following example illustrates that a subClassOf relationship among

devices is transitive.

```
(CleaningRobot rdfs:subClassOf Robot)
^(Robot rdfs:subClassOf Device) ->
(CleaningRobot rdfs:subClassOf Device)
```

Rule-based inference could provide forward chaining, backward chaining and hybrid models to execute rules over RDF graph. Here are example rules to show how robots' data is reasoned and how reasonor deduces an informed action. In these rules, a cleaning robot tests whether the light of room TS354 is on, and reads previous cleaning time from RFID tag. If the light is on, it may means that someone is working in the office, cleaning robot will go away. If previous cleaning time is not longer than 24 hours, room is no need to be cleaned. In case that the light is off and previous cleaning time is longer than 24 hours, cleaning robot will clean this room.

```
(roomTS354, lightIntensity, on) ->
(cleaningRobot521, goAway, roomTS354)

(rfidTS354, previousCleaningTime, ?PT)
^lessThan(currentTime-PT, 24hours) ->
(cleaningRobot521, goAway, roomTS354)

(roomTS354, lightIntensity, off)^
(rfidTS354, previousCleaningTime, ?PT)
^greaterThan(currentTime-PT, 24hours)
-> (cleaningRobot521, clean, roomTS354)
```

6 SIMULATOR

We develop a simulator to simulate the EN-based communication between devices and provide the semantic support from the swarm server. We create four types of devices in the simulator: cleaning robot, temperature sensor, a wireless device designed for reporting the patient's well-being to nurses (Riekki et al. 2007) and a location sensor. In the indoor robot system scenario we described in this paper, we pay attention to cleaning robot and temperature sensor in the simulator.

In the simulator, the user can specify messages for any device, assemble short packets or complete packets, and send them to the swarm server. The swarm server transforms packets into RDF format, adds them into context information ontology model, and reasons on data from the robots. The deduced result will be delivered to the specific device or all devices according to the entity identification of the deduced packet. We add one entity in front of each message to describe the message itself: the type, the sender, receiver, and a sequence number. This entity is included in template. In a real implementation, such an entity will not be needed if the same information is carried by the lower layer protocol.

Figure 4 shows situations in which the cleaning robot has sent a short packet to the swarm server and got responded action. In the cleaning robot window, user can specify the details of the message: the receiver's ID, message type, and the data content which shows the room situation that the cleaning robot is located in, including previous cleaning time and whether the room's light is on. Cleaning robot can assemble complete or short message, and will get corresponding message format from the swarm server. The swarm server responds the robots' message automatically, and sends the deduced message according to the rules specified in previous section.

The first screenshot in Figure 4 shows a short packet including the previous cleaning time and light on/off as following:

```
[urn:uuid:9c38ee "900"
"2008-05-26T05:00:00" "false"]
```

In this message, the identifier `urn:uuid:9c38ee` determines the template, `900` is the sequence number, and the last two values are the previous cleaning time of the room and the light off boolean tag. The swarm server deduces that the cleaning robot should clean this room TS354 and informs the cleaning robot to clean it by the following short packet:

```
[urn:uuid:bd61dee5-e9b8-422e-948d-
e7799caae04b "1001" EE#TS354 "pending"]
```

In this message, 1001 is sequence number, and the following value shows the room need to be cleaned and the action status is pending.

In the simulator, the swarm server and the robots can send 17 different packets in total: the server can send requests to the robots, while the robots can reply to the server's requests, and send asynchronous inform to the server. We calculate the length of these 17 packets, and find the result is quite promising. Short EN packet format contains on the average only 12.68% of the characters that a complete package contains and 9.67% of the characters that the corresponding RDF document contains.

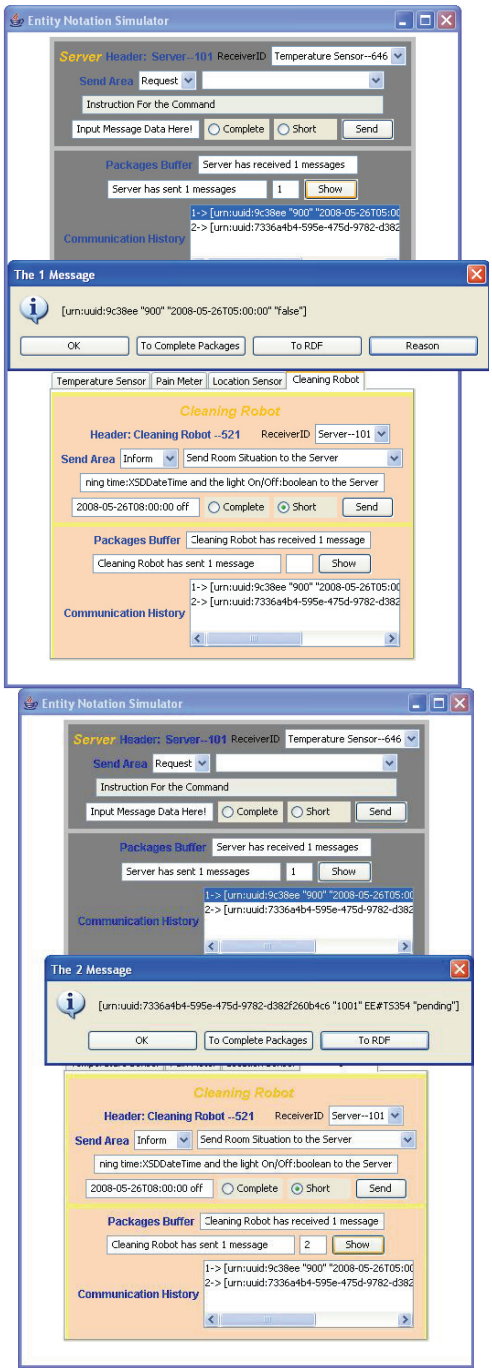


Figure 4: The Simulator Simulates the Cleaning Robot Scenario.

7 RELATED WORK

The idea of using Semantic Web technology to facilitate swarm intelligence has been reported by Boley (2007), but no resource-constrained robots have been integrated into the Semantic Web-based system before. We propose EN as a lightweight data representation for robots to support Semantic Web technology. We have compared EN in (Riekki, Su and Haverinen 2008), with XML compression technologies, e.g. Gzip (Gzip.org 2003), XMLPPM (Cheney 2000), XMLZip (XMLSolutions 1999), and some markup or data serialization languages, like JSON (json.org 1999), and YAML (yaml.org 2008). The result shows that the EN can almost reach the best compression ratio among XML compression technologies, while requiring minimal amount of computation for composing EN packets. Another alternative is binary XML formats, e.g. EXI (W3C.org 2008), WAP Binary XML Content Format (W3C.org 1999), and Infoset (W3C.org 2004). They are designed to provide a compact representation of XML for communication. They mainly use token-based compression schema in which each element, attribute, and so forth is encoded as a small integer value. Compared with them, EN could reach better compression rate and require less message sterilization processing power. Besides, EN has the advantage over the XML compression and binary XML solutions is that the EN packets are more human readable.

Many systems also adopted the solutions of expressing context information as ontology and using Description Logic (DL) based rules. Chen et al. (2004) proposed OWL to represent context information, and they utilized Java Expert System Shell (Jess) (Friedman-Hill 2008) to realize rule-based reasoning. Want et al. (2004) also modelled context by OWL in semantic Space project. They designed a two-layer context model for expressing context information, and reasoning was implemented using Jena. Korpipää et al. (2003) also presented a work on developing a lightweight ontology for mobile device context-awareness. But their context structure is represented using RDF, and this ontology contains only concepts, properties, and concept taxonomies, but no constraints. Our system has the advantage that the system can produce real-time deduced packets, and inform the robot swarm to perform them.

8 CONCLUSIONS

This paper presented an approach for semantic support of resource constrained robots in intelligent applications. We described our EN-based communicative architecture, which can complete the loop between robots and the swarm server by EN packets. We proposed EN formats, context

information ontology model, and a reasoner based on context information. We presented experiments and some promising results using a simulator.

The EN-based communicative architecture can be utilized for very resource-limited robots. It enables semantic functionality for the swarm of robots. The EN allows very simple and short packets to be sent by the robots; on the other hand these packets can be transformed into advanced knowledge representations in an unambiguous fashion. For example, when the robot does not even contain any OS but all functionality is programmed directly using C, it is straightforward to use UUID values as constants that are used to identify received packets and placed at the beginning of the sent packets.

The swarm server can transform EN packets from the robots into RDF format, use the information to deduce new tasks relevant to context information, and adapt the operation of the swarm to perform given actions efficiently. The context information ontology model and inference mechanism of swarm server provide semantic support for the swarm robots.

The simulation result shows that the length of short packets is less than 10% of the corresponding RDF document's length. These results are compared with other lightweight representation. Furthermore, when short EN packets are used, only composer and decomposer are needed for pre-process. A resource-constrained device does not need to run any complex algorithm to process the packet.

The future work includes building the first prototype where real robots use our EN-based communication. We will consider the uncertainty and dynamics in a real robot swarm system. EN can also be used in robot-to-robot communication in order to minimize bandwidth and computational overhead. Finally, we will develop the lightweight communication framework further based on this work, for example, to make more complex data structure be transferred to EN possible.

ACKNOWLEDGEMENTS

This work was funded by Infotech Oulu. The author would like to thank the participants of ROBOSWARM project.

REFERENCES

Boley, H & Chang, E 2007. Digital Ecosystems: Principles and Semantics, in *Proceeding of IEEE International Conference on Digital Ecosystems and Technologies*. IEEE Computer Society Press, Cairns, pp. 398-403.

Chen, H, Finin, T, Anupam, J, Kagal, L, Perich F & Dipanjan, C 2004, 'Intelligent Agents meet the Semantic Web in Smart Spaces', *Internet Computing*, vol. 8, no. 6, pp. 1545-1651.

Cheney, J 2000. Compressing XML with Multiplexed Hierarchical PPM Models, in *Proceedings of the IEEE Data Compression Conference*. IEEE Computer Society Press, Snowbird, pp.163-172.

Dey, A & Abowd, G 2000. Towards a Better Understanding of Context and Context-Awareness, in *Workshop the what, who, where, when, and how of context-awareness at CHI 2000*. ACM Press.

Friedman-Hill, E 2008, Jess, the rule engine for the javaTM platform, viewed by 24 April 2009, <http://herzberg.ca.sandia.gov/jess/index.shtml>.

Gzip.org 2003, *The gzip home page*, viewed 24 April 2009, <http://www.gzip.org/>.

Holland, O & Melhuish, C 1999. 'Stimergy, Self-Organization, and Sorting in Collective Robots', *Artificial Life*, vol.5, no. 2, pp. 375-393.

Json.org 1999, *Introducing JSON*, viewed 24 April 2009, <http://www.json.org/>.

Korpipää, P, Mantyjärvi, J, Kela, J, Keranen, H & Malm, EJ 2003. 'Managing Context Information in Mobile Devices', *Pervasive Computing*. vol.2, no. 3, pp. 42-51.

Riekki, J, Alakärppä, I, Koukkula, R, Angeria, J, Brockman, M & Saloranta, T 2007, Wireless Pain Monitoring, *in The 2nd International Symposium on Medical Information and Communication Technology*. University of OULU Press.

Riekki, J, Su, X & Haverinen, J 2008. Connecting Resource-Constrained Robots to Knowledge-Based System, in *Proceeding of International Conference on Modelling, Identification and Control*. ACTA Press.

Studer, R, Benjamins, VR & Fensel, D 1998. 'Knowledge engineering: principles and methods', *Data Knowledge Engineering*, vol.25, no.1-2, pp. 161-198.

Wang, X, Dong, JS, Chin, CY, Hettiarachchi, SR & Zhang, D 2004, 'Semantic Space: An Infrastructure for Smart Spaces', *Pervasive Computing*, vol.3, no. 3, pp. 32-39.

W3C.org 2004, *RDF Primer*, viewed 24 April 2009, <http://www.w3.org/TR/REC-rdf-syntax/>.

W3C.org 2008, *Efficient XML Interchange (EXI) Format 1.0*, viewed 24 April 2009, http://www.w3.org/TR/2008/WD-exi-20080919/.

W3C.org 1999, *WAP Binary XML Content Format*, viewed 24 April 2009, <http://www.w3.org/TR/wbxml/>.

W3C.org 2004, XML Information Set (Second Edition), viewed by 24 April 2009, <http://www.w3.org/TR/xml-infoset/>.

XMLSolutions 1999, *XMLZip - XML Solutions*, viewed 07 June 2008, <http://www.xmls.com/>.

Yaml.org 2008, *YAML Ain't Markup Language Version 1.2*, viewed 24 April 2009, <http://yaml.org/spec/1.2/>.

TUNING OF INDUSTRIAL CONTROLLERS OVER PUBLIC IP NETWORKS

Renato F. Fernandes Jr, Dennis Brandão
University of São Paulo, Av.Trabalhador Sãocarlense 400, São Carlos, SP, Brazil
renfernand@yahoo.com.br, dennis@sc.usp.br

Nunzio M. Torrisi
Universidade Federal do ABC, Rua Santa Adélia 166, Santo André, SP, Brazil
nunzio.torrisi@ufabc.edu.br

Keywords: PID control, Identification, Industrial systems, Remote control, Industraial networks.

Abstract: Tuning of industrial systems is executed in the initial phase of the system and mainly during the maintenance phase, providing characteristics of the performance of the industrial process during production life cycle. Remote tuning supports several practical applications, such as specialized companies outsourcing services or companies distributed in different areas centralizing optimization. This paper proposes a software tool for remote tuning of open or closed PID control loops in an industrial environment that fulfils the requirements described above, in a single platform. The software tool could be used in control loops tuning in industrial systems, as well as in an academic environment simulating control applications and industrial networks.

1 INTRODUCTION

Increasing competitiveness in the industrial sector has required continuous improvement in product quality processes, optimizing the production and reducing operational costs. Nowadays, researches like (Avov, 2004) show the potential of the Internet in the industrial environment. However, considering the use of the Internet for control and supervision of industrial processes, it should be noticed that the nature of production and automation systems demands some requirements to be secured, such as managing multiple accesses, ensuring the communication and the control system, setting maximum periods for process data updates and quality of service maintenance (Abdelzaher, 2002).

Remote access architectures may be implemented at different levels in the control hierarchy: at process level, at supervisory level and at system optimization level (Yang, 2003). In terms of process, the proposal is to include remote control within the process control loop, according to (Overstreet, 1999) (Yang, 2007). For this purpose, the conventional discrete control structure should be altered to conform to the Internet's variable (Luo, 2000), (Yang, 2007).

The concern at the supervisory level is safety and quality of service, discussed in (Kunes, 2001). Yang et al. (Yang, 2003) proposes a remote control at supervision level for services that are independent of the Internet delay, which would be restricted to acyclic services such as tuning parameters for PID block and set points.

In the context of SCADA (Supervisory Control and Data Acquisition) systems, the OPC (OLE for process control) technology (OPC Foundation, 2006) combined with Web technologies, as WebServices, is used to draw complex architecture for manufacturing in order to create communication system directly between shop-floor and decentralized supervision systems (Zheng and Nakagawa, 2001).

Commercial companies offer today solutions for remote monitoring and tuning of industrial systems. However, these solutions have some limitations, because they may be based on non-standard platforms for the industrial environment and may use common WebServices (Calvo, 2006) (Batur, 2000) (Qin, 2007).

Torrisi (Torrisi, 2007) proposes a standard OPC communication mechanism based on the Internet, a platform-independent alternative to WebServices. That standard, called CyberOPC, uses "open"

security technologies for light software components, therefore providing better performance and increasing guarantees for data security, when compared to other technologies based on WebServices.

This paper proposes an architecture to execute remote tuning of industrial control systems using the Internet, fulfilling acceptable security and performance requirements. In order to validate the architecture, a software application using CyberOPC and called Cybertune will be presented. The validation consists of a model-based identification and tuning using first-order-plus-dead-time systems.

This paper is organized as follows: Section 2 shows the requirements for remote tuning and problems related to supervision and remote control of industrial systems. Section 3 presents the generic architecture of the remote tuning system. Section 4 describes tests and results from remote tuning, and finally section 5 presents the conclusions and indicates future researches.

2 REMOTE TUNING REQUIREMENTS

In control loop tuning, the system identification phase demands over half of the effort (Yu, 2006), (Hjalmarsson, 2005). In order to obtain the model in time domain, the most common way to identify low order systems is identifying the transient response to a process alteration. The transient response is found when stimulating the system using a step-format signal, an impulse or a pseudo-random binary sequence (PRBS) as a system input. Identification is executed by estimating parameters through data collection and expression using ARX (Auto-Regressive Exogenous) or ARMAX (Auto-Regressive Moving Exogenous) model (Aguirre, 2004) (Ljung, 1999).

Controllers are tuned in three phases, according to model-based techniques: identifying the plant model, tuning based on the identified model, and validating tuning using simulations based on the identified model with the new controller.

An important step in the model-based identification consists of defining the sampling rate, which must be constant and generally in the order of ten times smaller than the time constant of the system (Aguirre, 2004).

In general, the OPC technology has an update rate defined in seconds for processes with fieldbus technology. This rate reduces our scope to slower systems with time constants and dead time 5 to 10 times of the time of the OPC acquisition, although many industrial systems may fit this scenario, such as temperature control and chemical processes.

2.1 Communication Problems related to Remote Control and Monitoring

The Internet and WebServices available nowadays create some obstacles when used in industrial control systems (Abdelzaher, 2002), (Yang, 2003), (Torrisi, 2007):

- Communication delay of different types, throughout the data source up to the destination nodes.
- Non-determinism of the network due to various routes available on the Internet, where the decision of the best route to be used should be taken for each data packet received.
- Network data should be secured, meeting the following security properties: confidentiality, authentication and integrity of messages.
- Considering the remote control in public networks, OPC solutions based on WebServices are even slower than those based on DCOM(Distributed Component Object Model) (Advosol Inc., 2004).

3 GENERIC ARCHITECTURE FOR REMOTE TUNING

The architecture proposed for the remote tuner is based on the interconnection of modules in three different contexts: the industrial plant, the server and the client. It is based on the client-server cooperation model, which consists of different interconnected modules providing process and configuration variables from the industrial plant to the remote client. The proposed architecture is implemented in a generic "open" design and can be used with any commercial software component. The figure below shows the components of the architecture.

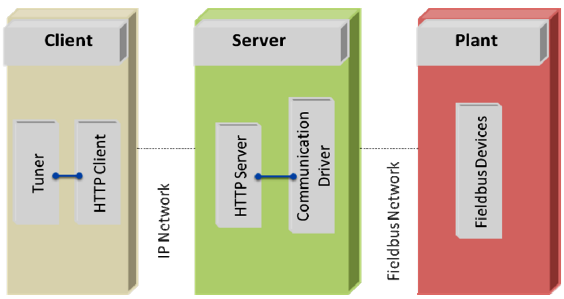


Figure 1: Remote Tuner Architecture.

Inside the "server" module, indicated in the architecture proposed in Figure 1, the communication driver is the communication interface between the server and field devices for data acquisition. Being a widely used standard in the industrial environment, OPC was selected for this project to communicate to the fieldbus network.

Moreover, in the server module, the HTTP (Hypertext Transfer Protocol) Server is responsible for processing remote requests from several clients connected to server module. The communication between the remote HTTP server and the clients can use proprietary protocols, WebService and other protocols over HTTP.

The "client" module represents the remote monitoring and tuning unit. The client must be a standard OPC client allowing the communication to a variety of field device networks. The Cybertune prototype for this study used an OPC client architecture for local communication and a CyberOPC client for remote communication.

In order to avoid the non-determinism of the Internet, the strategy used in this work was attaching a timestamp to each sample. This way, the time measured will always be used, thereby obtaining the actual rate for each performed identification. For the remote communication, using CyberOPC ensures that there will be a constant data acquisition on the control side, with packet sequencing, determining the information obtained by the remote client.

CyberOPC is based on open standard technologies. It minimizes software and interfaces layers for a better and faster network use because it is a technology with a simple philosophy dedicated to industrial applications with "soft real time" requirements in IP networks with assured bandwidth. One of the features from this protocol is the use of an internal cache memory, which provides better performance in processing messages.

Server and client roles in the proposed architecture using CyberOPC communication protocol are detailed below.

3.1 The CyberOPC Communication System

The most commonly used OPC specification is the OPC Data Access (DA) 2.x and 3.x. It does not intend to compensate communication latencies that can occur in wide area networks neither provide communication for clients or servers intermittently connected to the network. Another restriction comes from the fact that OPC DA data cannot be transferred over the Internet, because all TCP/IP ports except Port 80, which allows only textual data, are usually blocked at corporate firewalls. However, OPC is based on a Microsoft standard for component communication – DCOM (Distributed Component Object Model) – that is not textual.

The OPC XML-DA and the incoming OPC-UA specification (OPC Foundation, 2008) define a WebServices based approach for reading from and writing data to plant floor automation systems. In this approach, all OPC data are formatted in XML blocks and transmitted using simple object access protocol (HTTP-SOAP). The choice to adopt WebServices solves the problems related to binary data blocked at firewalls and promotes software integration manufacturing with WebServices technologies, despite the low quality of service level to address typical timing requirements of real time process control networks.

Step 1 in Figure 2 represents the processing of CyberOPC commands without SOAP preprocessor. Introducing the OPC cache strongly reduces calls to the OPC Client. Tests have reported the reduction of 70% of the Message Broker Time when compared to the time consumed by a Gateway WebServices based. Steps 2, 3 and 4 represent the interaction between the OPC library and the CyberOPC Application Server

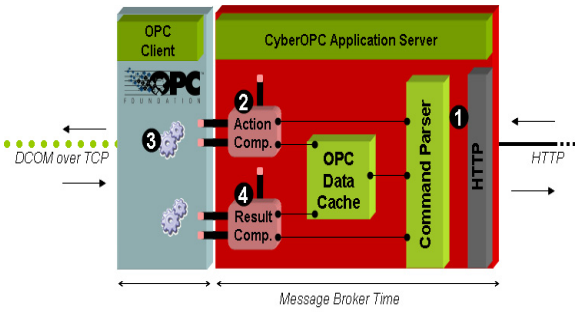


Figure 2: Message Broker Time for CyberOPC Gateway. OPC Packet Data and requests are encoded as text using JSON or XML syntax

For HTTP communications, there are two categories of security mechanisms: transport level security and message level security. The transport level security mechanism uses Server Secure Socket (SSL), using digital X.509 certificates (ITU Recommendation, 1997). The CyberOPC communication system replaces the OPC Polling mechanism over SSL.

3.2 The Cybertune Structure

The Cybertune prototype consists of four main

operational modules: data acquisition, system identification, model transformation and tuning modules. A schematic of the structure is shown in figure 3.

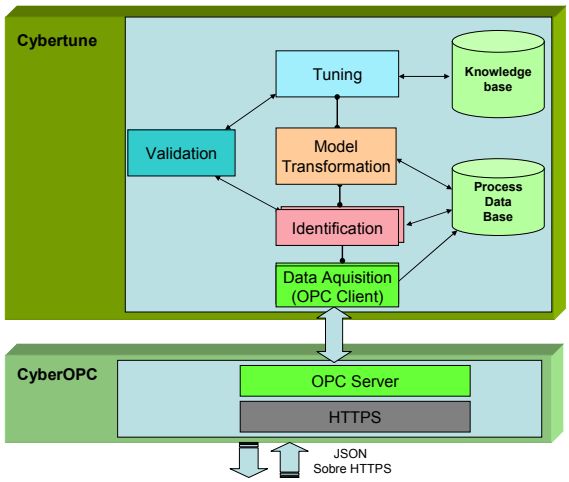


Figure 3: Cybertune structure.

The data acquisition module consists of an OPC or CyberOPC client in accordance with the specifications of the OPC standard (OPC Foundation, 2008) or the specifications in (Torrisi, 2007). Thus, the component interface has the same data access philosophy, consisting of OPC library records, adding groups and items to the database, and acyclic communication per event, where the client is notified when a new data event is issued by the server.

For the identification module, responsible for determining the system transfer function, the ARX model was used, which provides good results for first and second order linear systems, the most common systems in industrial environments (Yu, 2006). As this project aims to validate the architecture for online identification and tuning, the Cybertune needs to receive process data and automatically perform the identification.

Assuming that there are communication delays and transmission failures while executing online remote identification, this work proposes the following methodology. First, every sample collected by CyberOPC has a timestamp with the time when the data was acquired by the gateway. In addition, since the ARX model requires continuous sampling and CyberOPC sends data (per event on data change) in an optimized way, it is necessary to rebuild the process signal at a constant sampling rate.

To solve this case, a "pre-identification" module was included, being responsible for receiving queued data from the acquisition module and sending samples to the identification data queue at a constant sampling rate. Interpolation between two sample points used the first order equation. An example of this architecture is shown in figure 4.

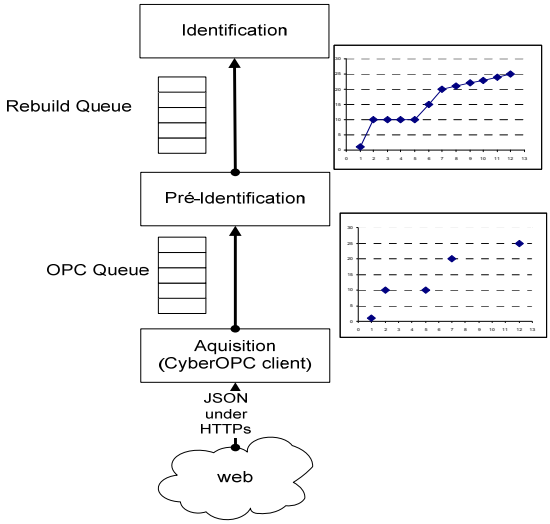


Figure 4: Pre-Identification module.

After the identification and validation of the ARX model is necessary to transform the ARX model in open loop model to be subsequently applied to the model based methods for tuning. The algorithm used for model transformation module for open and close loop was described in (Fernandes and Brandão, 2008).

Finally, the tuning module applies the tuning methods to the model obtained before. The Cybertune uses Ziegler Nichols (ZN) and internal model control (IMC) common model-based tuning methods (Ang, 2005).

4 TESTS AND RESULTS

Identification tests were performed to validate the proposed architecture simulating first-order-plus-dead-time systems using local and remote identification in a corporate network.

Tests were conducted using the FieldBus Plant Simulator (FBSIMU), which simulates industrial plant and fieldbus control logic. .The studies of (Pinotti and Brandão, 2005) showed that FBSIMU has good approach to simulate real system.

Figure 5 presents the tests scenarios of this work. Local tests used communication with Cybertune and FBSIMU in the same station. Remote tests used a 2Mbits internet connection.

Server station consists of a CyberOPC gateway module communicating via OPC to FBSIMU. Remote client station consists of Cybertune communicating via CyberOPC protocol with the FBSIMU.

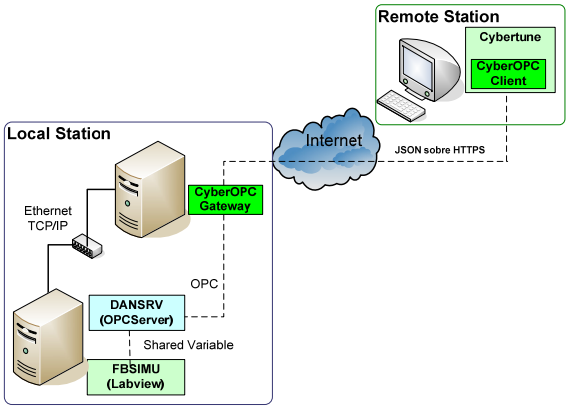


Figure 5: System architecture for local and remote communication between Cybertune and FBSIMU.

Simulation tests were performed using three systems with different characteristics: two tests used slower systems (such as an oven or industrial chemical process), and the other test used a fast system (such as a flow control loop). The transfer function of the systems were showed below, where in this paper is called system 1 the equation (1), system 2 the equation (2) and so on.

$$G_p = \frac{2}{100\,s+1} e^{-50\,s} \tag{1}$$

$$G_p = \frac{25}{15\,s+1} e^{-35s} \tag{2}$$

$$G_p = \frac{3}{150s+1} e^{-20\,s} \tag{3}$$

To validate the tests, the ITAE performance index and the correlation index (FIT) were used in relation to the actual signal and the identified signal. For a correlation index higher the identification is considered as good (Ljung, 1999).

3.1 Tests Results

For the first test, consider the transfer function (1) of the system 1.

Initially regarding the local identification test, identification is estimated according to approximation using a fourth order ARX model and sampling rate (To = 1.0 sec). The model shown in (4) was obtained with FIT=96.75%:

$$FTMA(z) = \frac{-0.0170z^3 + 0.0060z^2 + 0.0105z + 0.0023}{z^4 - 1.0030z^3 - 0.4280z^2 - 0.0129z + 0.4448} \tag{4}$$

Then, the ARX model is transformed into the open loop model according to the equations proposed in (Fernandes e Brandão, 2008), which results in the following model approximation:

$$G_p = \frac{2.00}{102.3\,s+1} e^{(-46.30\,s)} \tag{5}$$

The graph shown in Figure 6 compares the real system and the system identified locally. The final solution has FIT equals to 98.21%.

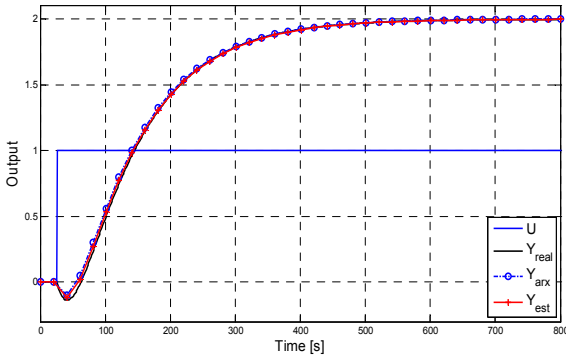


Figure 6: Cybertune identification of system 1 in a local station. It shows the original signal (Yreal), the fourth order ARX signal (Yarx) and the identified open loop system (Yest).

The same system defined in (1) was used in the remote identification test. As described in the previous test, considering an approximation ARX with a fourth order and sampling rate (To = 2.0 sec), the model is estimated according to the following equation (FIT=93.78%):

$$FTMA(z) = \frac{-0.0028z^3 - 0.0120z^2 + 0.0083z + 0.0080}{z^4 - 1.4310z^3 + 0.1628z^2 + 0.07639z + 0.1931} \tag{6}$$

After transforming the ARX model into the open loop model, the model approximation is obtained:

$$G_p = \frac{2.00}{118.7\,s+1} e^{(-38.7\,s)} \tag{7}$$

The graph shown in Figure 7 compares the real system and the system identified remotely. The final solution has FIT equals to 93.65%.

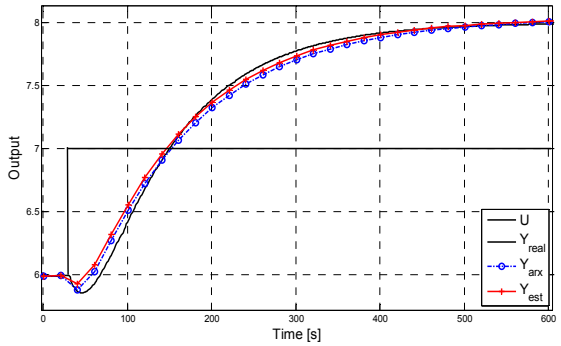


Figure 7: Cybertune identification of a system 1 in a remote station. It shows the original signal (Yreal), the fourth order ARX signal (Yarx) and the identified open loop system (Yest).

Table 1 summarizes the results from the three tested systems. The experimental procedure for systems 2 and 3 are omitted because they are the same as described for system 1.

Table 1: Local and Remote Tests Results.

System	Description of the test with Cybertune	ITAE	FIT[%]
1	Local identification (T_o = 1.0 sec)	4.57E+2	98.21
	Remote identification (T_o = 2.0 sec)	6.34E+3	93.65
2	Local identification (T_o = 1 sec)	5.93E+2	98.46
	Remote identification (T_o = 5 sec)	8.13E+3	91.60
3	Local identification (T_o = 1 sec)	3.54E+2	99.77
	Remote identification (T_o = 5 sec)	1.86E+3	92.80

In the tuning phase is used the open loop model obtained from the identification phase. The figure below shows the tuning with common methods ISTE, ITAE and IMC and ZN.

5 CONCLUSIONS

This paper proposed an architecture to execute remote tuning of industrial control systems using the Internet, fulfilling acceptable security and performance requirements. In order to validate the architecture, a software application using CyberOPC and called Cybertune were presented. The validation

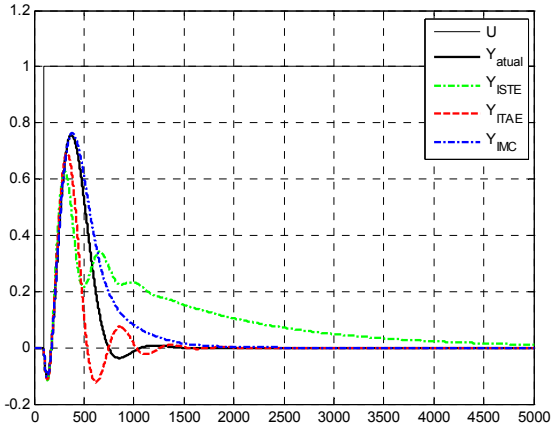


Figure 8: Example of tuning the model obtained in (7) in a remote station. It shows the original tuning with ZN (Y_{atual}) and some common methods based in model.

consisted of a model-based identification and tuning of three selected first order plus dead time systems once this is a typical class of industrial systems, but the architecture can be extended to other configurations.

The tests and results session of this paper focused on a given first order system with dead time for PID controllers, tests for other two systems were conducted and the results were summarized.

The tests demonstrated that the remote model identification is very close to the local identification and the original system, which validates the architecture for identification and subsequent tuning implemented with model-based methods.

For remote identification, it is necessary to pre-filter the signal in order to increase the efficiency of ARX identification.

In future researches, we intend to validate the algorithm in real plant floor systems, through fieldbus system applications.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Brazilian agency FAPESP for financial support received and the academic support and research structure of the Engineering School of São Carlos - University of São Paulo.

REFERENCES

Abdelzaher, T. F..; Shin, K.G.S.; Bhatti, N., 2002. Performance Guarantees for Web Server End-Systems:

A Control-Theoretical Approach, In: *IEEE Transactions on Paralllel and distributed systems*, vol 13, no 1, January 2002.

Aguirre, L. A., 2004. Introdução a Identificação de Sistemas, Técnicas Lineares e Não Lineares aplicadas a sistemas reais. 2ª edição, 2004, Editora UFMG.

Ang, K. H.; Chong, G.; Li, Y. (2005). PID Control System Analysis, Design, and Technology. In: *IEEE Transaction on Control Systems Technology*, Vol 13, No 4, July 2005.

Avoy, T. M.; Jounela, S.L.J.; Patton, R.; Perrier, M.; Weber, H.; Georgakis, C., 2004. Milestone report for area 7 industrial applications. In: *Control Engineering Practice* 12 (2004) 113-119.

Advosol Inc., XMLDA.NET White Paper, 2004 Available:*http://www.advosol.com/driver.aspx*?Topic= WhitePaperXMLDANET.

Batur, C.; Ma, Q.; Larson, K., Kettenbauer, N., 2000. Remote tuning of a PID position controller via internet. In: *American Control Conference*, Chicago, 2000.

Brandão, D., Cunha, M. J., Pinotti, J. M., 2004 . Fieldbus Control System Project Support Tool based on Experimental Analysis and Modeling of Communication Bus. In: *IEEE International Conference on Industrial Technology*, 2004, Hammamet, 2004.

Calvo, I.; Marcos, M.; Orive, D.; Sarachaga, I., 2006. A methodology based on distributed object-oriented technologies for providing remote access to industrial plants. In: *Control Engineering Practice,* 14 (2006), pp. 975-990

Fernandes, R. F., Brandão, D., 2008. Método de identificação online de sistemas industriais com controladores PI em malha fechada, In: *Conferência Internacional de Aplicações Industriais*, Poços de Caldas. Anais do VIII INDUSCON, 2008.

Hjalmarsson, H., 2005. From experiment design to closed loop control. *Automatica,* 41 (2005) pp. 393-438.

ITU Recommendation, 1997. X.509 version 3. Information Technology - Open Systems Interconnection - The Directory Authentication Framework, August 1997.

Ljung, L., 1999. *System Identification – Theory for the User*. Second Edition, 1999, Englewood, Prentice Hall.

Luo, R.C.; Chen, T.M., 2000. Development of a multibehavior-based mobile robot for remote supervisory control through the Internet. In: *IEEE/ASME Transactions on mechatronics*, vol. 5, no. 4, December 2000.

Kunes, M., Sauter, T., 2001. Fieldbus-Internet connectivity: The SNMP approach. In: *IEEE Transactions on Industrial Electronics*, 48(6), 1248–1256.

Kurose, J. F., Ross, K. W., 2006. *Redes de computadores e a Internet*. 3ª Edição, 2006, Editora Addison Wesley.

Qin, W., Wang, Q., 2007. An LPV approximation for admission control of an internet web server: Identification and control. In: *Control Engineering Practice*, Feb 2007.

OPC Foundation, 2006. OPC Standard. In: <*www.opcfoundation.org*>. Access em: 26 jul. 2007.

Overstreet, J. W., Tzes, A., 1999. Internet-based client/server virtual instrument designs for real-time remote-access control engineering laboratory. In *Proceedings of the American control conference*, Vol. 2, pp. 1472–1476.

Pinotti Jr., M., Brandão, D., 2005. A flexible fieldbus simulation platform for distributed control systems laboratory courses. *The International Journal Of Engineering Education*, Dublin, v. 21, n. 6, p. 1050-1058, 2005.

Torrisi, N. M. ;Oliveira, J. F. G. 2007. Remote Control of CNC Machines using the cyberopc communication system over public networks. *International Journal of Advanced Manufacturing Technology*, v. 2007, p. 001, 2007

Torrisi, N. M., 2007. Sistemas de comunicação para redes IP dedicado a dados industriais codificados em JSON. Patente submetida ao Instituto Nacional da Propriedade Industrial (INPI), sn 018070047508, 2007.

Yang, S.H.; Chen, X.; Alty, J.L., 2003. Design issues and implementation of internet-based process control systems. In: *Control Engineering Practice* (11) 2003, 709-720.

Yang, S. H., Dai, C., Knott, R.P., 2007. Remote Maintenance of Control System performance over the Internet, In: *Control Engineering Practice*, Volume 15, Issue 5, May 2007, Pages 533-544

Yu, C. C., 2006. *Autotuning of PID Controllers: A Relay Feedback Approach*. 2nd Edition, Springer, 2006.

Zheng, L., Nakagawa, H., 2002. OPC (OLE for process control) specification and its developments, SICE 2002 – *Proceedings of the 41st SICE Annual Conference*, pp. 917-920 vol. 2, 2002.

DYNAMIC CONTROL OF NETWORK PROTOCOLS
A New Vision for Future Self-organising Networks

Sven Tomforde, Emre Cakar and Jörg Hähner
Institute of Systems Engineering, Leibniz Universität Hannover, Appelstr. 4, 30167 Hannover, Germany
{*tomforde, cakar, haehner*}*@sra.uni-hannover.de*

Keywords: Organic Computing, Data communication network, Protocol, Parameter, Optimisation.

Abstract: In recent years communication protocols have shown an increasing complexity, in particular in terms of the number of variable parameters. Data communication networks like, e. g. , the Internet reach the limits of their extensibility which leads to initiatives coping with the future of the Internet and data communication in general. A first step towards creating a sustainable solution without exchanging the whole system is to make the static character of network protocols more flexible. An adaptive behaviour of nodes within a network and an autonomous, self-organising concept for their control strategies leads to a possible increase in performance accompanied by an increase of extensibility. This paper presents a new vision of how to establish these new control strategies mostly independent of the particular protocol by using the concepts of Organic and Autonomic Computing. We introduce an adaptive and automated network control system for the dynamic and self-organised control of protocol parameters. This system consists of two sub-systems: an on-line adaptation mechanism and an off-line learning component. The current status is introduced in combination with the definition of further challenges and fields of research.

1 INTRODUCTION

Recent years were characterised by a dramatical growth of communication need and increase of traffic over data communication networks. In combination with the ascending number of protocols and their varying configuration possibilities (configuration space) the complexity of the control task at each node in the network is growing. Based on this observation, more and more often the question arises whether the current structure of the network (in particular the Internet) will be able to cope with the increasing demand (cf. e. g. (Handley, 2006)). This leads to a new vision for the future of the Internet ((Siekkinen et al., 2007)).

The number of researchers formulating the need to exchange the complete set of techniques (e. g. protocols, structure, etc.) is increasing steadily. One major problem focused here is that the existing protocols are designed as static solutions. Although the situation at particular nodes within the communication networks (in terms of e. g. resource usage, available bandwidth, currently known neighbours, etc.) changes over time, the configuration is typically not adapted to the current requirements. A possibility to solve this problem by keeping downward compatibility (this means cooperation of static and dynamic solutions) is presented within this paper.

Based on the approaches of Organic Computing (OC - cf. (Schmeck, 2005)) and Autonomic Computing (AC - cf. (Kephart and Chess, 2003)) an adaptive network control system is introduced which aims at coping with the large configuration space. The system is locally organised, adaptive, and has learning abilities guaranteeing the best possible performance for each node.

This paper presents an adaptive and automated system for the dynamic and self-organised control of network protocol parameters (e. g. values for timeouts, maximum number of re-transmissions, number of open connections, etc.). Section 2 focuses on a short overview of already introduced approaches to optimise network settings and concludes with the statement that no system with the required properties exists yet. In Section 3 the architecture and general approach for the proposed system are described combined with details on the technical realisation. Afterwards, Section 4 defines the challenges for the further development and the research to be done until a real-time operation can be applied. Finally, Section 5 concludes the vision for dynamic network control.

2 STATE OF THE ART

The dynamic selection of network protocol parameter settings depends on the situation-based generation of these settings. Therefore, the task can be divided into two different subtasks: the off-line optimisation of parameter settings for a given (observed) situation and the on-line adaptation of the network controller settings with a suitable parameter set.

The optimisation of parameter settings deals with the problem to determine a set of parameters for a given protocol that is as close to the optimum as possible. The task is characterised by the required amount of time and the quality of the solution to be found. In this context *off-line* means evaluating new possible settings using simulation and thus without interfering with the live-system.

There are several examples where authors optimised the settings of their particular protocols, but their intention has been to optimise a specific protocol and not to create a generic system. Considering the techniques used in our system, the approach of Montana and Redi is connected as they also use an Evolutionary Algorithm (EA) to optimise a full custom communication protocol for military MANETs (Montana and Redi, 2005). A similar optimisation of a protocol (for underwater communications) using an EA is described by Sözer et al. (Sözer et al., 2000). Turgut et al. discuss the usage of a Genetic Algorithm to optimise their MANET-based clustering protocol in (Turgut et al., 2002). They all compare their achieved results to a manual optimisation. In contrast to the network control system presented in this paper the approaches are specific to the particular protocols, but do not aim at providing a generic system which is adaptable to different protocol types.

Due to the time-intensive process of generating optimal parameter sets an on-line usage in live-systems is not applicable. Hence, such a solution has to somehow combine the strengths of optimisation techniques with approaches to immediately react on an observed stimulus. Although research communities are aware of the demand and it already has been part of the vision of initiatives (Kephart and Chess, 2003) a solution has not been presented yet.

One approach towards a possible solution has been described by Ye and Kalyanaraman (Ye and Kalyanaraman, 2001). They introduced an adaptive random search algorithm, which tries to combine the stochastic advantages of pure random search algorithms with threshold-based knowledge about extending the search. Their approach is based on the initial system as presented in (Ye et al., 2001). In contrast to our approach, Ye et al. propose a centralised system that tackles the optimisation task for each node. To allow for such a division of work between a central server and the particular network nodes they have to deal with problems like e. g. bandwidth usage, single point of failure, or local knowledge accessible from server-side.

3 SYSTEM

The motivation to develop a dynamically adapting system has been formulated before (cf. AC (Kephart and Chess, 2003) or Autonomic Networking (Jennings et al., 2007)), a proof of concept is still missing. The system presented in this paper is a first step towards a possible realisation. Based upon our architecture as pictured in Fig. 1 and initially presented in (Tomforde et al., 2009) the responsibilities of parameter set generation and on-line adaptation of the control system are assigned to different layers. One component (Layer 2) evolves new parameter sets not being restricted by real-time requirements for time and computation power. The other part reacts on changing stimuli (observed situation). This division of functionalities leads to the possibility that for an observed situation no matching optimised parameter set is available. In this case a covering mechanism has to cope with the situation which chooses the best possible control and adaptation strategy.

Technically, the architecture is realised by two connected techniques. As described in e. g. (Schmid et al., 2006), a combination of evolution and learning seems to be a promising solution to realise dynamic system-adaptations. Based on this assumption, for the off-line part an Evolutionary Algorithm (EA) is used in combination with a standard network simulation tool. The on-line learning mechanism is based on a modified version of Wilson's Learning Classifier System XCS (Wilson, 1995).

Within this Section the goals of the system are defined, followed by a short introduction of the basic concepts. The main part is dealing with the architecture and the current status.

3.1 Goal Definition

The network control system as presented in this paper aims at increasing the performance of data communication. It allows for the dynamic adaptation of network protocols to a continuously changing environment. Based on the initially introduced concept (Tomforde et al., 2009), the system requires *organic* (in terms of OC) characteristics – a decentral, self-organised approach leads to a stable, reliable con-

trol, the system is able to learn and optimise its behaviour autonomously. The network control system is generic, which means the controlled network protocol client can be exchanged. It supports a large set of different protocol types (Peer-to-Peer, mobile ad-hoc, wire-based, sensor, etc.) and protocols (e. g. BitTorrent (Cohen, 2003), Hyper-Gossiping (Khelil et al., 2007), etc.). If possible, the autonomous network control systems can collaborate and fullfill system-wide goals by using local interactions. The goal definition has some similarities with the manifest of Autonomic Computing (Kephart and Chess, 2003) and the Autonomic Management of Networks approach as presented in (Jennings et al., 2007). In contrast to the network-wide approach (which is unfeasible for large networks like the Internet), we assume that a locally organised solution based on local rules and local interactions will converge to a system-wide optimisation in most of the cases and therefore does not need global knowledge and global control.

3.2 Basic Concepts

The architecture is based on two already known approaches: the *Generic Observer/Controller Architecture* (Richter et al., 2006) and the *2-layered Architecture* of the Organic Traffic Control (OTC) System (Prothmann et al., 2008). The generic architecture describes an approach, where a decentralised system (System under Observation/Control (SuOC)) is wrapped with an additional surveillance and feedback mechanism. Sensors and actuators are used to monitor and control the SuOC by establishing a control loop. This control loop observes the behaviour of the SuOC through sensors, compares the results with expected behaviour and the current goals of the system, decides what action is necessary and controls the SuOC with the best known action through actuators. Additionally, a memory function keeps track of historical situations and control actions to be able to optimise the behaviour from existing knowledge.

The architecture of the OTC system is based on this approach. The realisation of a real-world scenario (control of traffic lights at urban intersections) leads to some restrictions – the main aspect is that the system has to use only parameter sets with guaranteed performance. Therefore, the situation-dependent creation of new parameter sets has been assigned to a new layer within the architecture where a simulation-based approach is performed off-line. This concept can be also found in the network control system, but the different domain leads to some modifications and changes, which will be explained in the remainder of this section.

3.3 Architecture

Similar to the *2-layered Architecture* of the Organic Traffic Control (OTC) System (Prothmann et al., 2008) our architecture consists of three parts: the SuOC at Layer 0, an on-line adaptation mechanism at Layer 1, and an off-line learning component at Layer 2 (see Fig. 1). All three layers will be presented in the following.

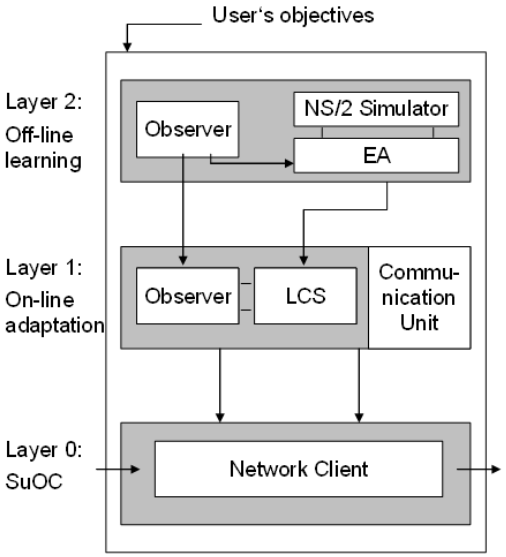


Figure 1: System architecture.

Layer 0: System under Observation and Control

The SuOC is a parametrisable Network Controller. Due to the generic concept of the proposed system it is not restricted to a particular set of protocols – the only restriction applied is that it has to provide a set of variable parameters and a local quality criterion (e. g. duration of a download in Peer-to-Peer systems or a weighted trade-off between energy consumption and broadcast-covering for MANETs) for the performance measurement. This means, protocols on all layers (media access to application) can be controlled in the same way as e. g. wire-based protocols or mobile ad hoc networks. A good setup of the variable parameters that match the current condition at the network node has an important influence on the resulting performance for these systems. In the architecture, the parameter setup is optimised on-line by the O/C component in Layer 1.

Layer 1: On-line Adaptation

The Layer 1 component can be divided into two different parts: an Observer and a Controller. The Ob-

server is responsible for monitoring the situation at a particular node. It measures those attributes having influence on the selection of appropriate parameters for the control strategy. This selection depends on the specific controlled protocol and typically contains attributes like buffer sizes, delay times, etc. Additionally, protocol-specific parameters like e. g. number of nodes in sending distance for MANET protocols or available system resources like CPU, upload-bandwidth, download-bandwidth, etc. for P2P protocols can be taken into account. Afterwards, these values are aggregated to an abstract situation description realised as an n-dimensional vector with n equal to the number of observed values.

The main part of the Controller is a Learning Classifier System, which is based on Wilson's XCS as introduced in (Wilson, 1995). The LCS maps the aggregated input-information from the Observer to a rule base of possible actions, the process is realised in accordance with Wilson's approach. The basic change in concept is that our LCS version is not able to create new rules as this process can lead to unwanted behaviour (random rule generation). This leads to the problem that the system might not have a matching rule for the currently observed situation, although the system detects the demand to adapt the network client. Therefore, a covering mechanism is needed, which chooses the best possible action.

This covering is realised based on the assumption that a classifier whose condition part is located close to the current situation description – although it does not match it – is better than any other one existing within the rule set. Due to this assumption a covering process is executed which selects the "nearest" classifier in terms of the Euclidian Distance calculated for the n dimensional vectors (equal to the situation description) and using the centroids of the intervals used for each interval predicate. This classifier is copied, its condition part is adapted to the current situation description (using a standard interval size around the given situation), and it is added to the rule set. Based on this simple process we ensure to only use tested actions and we also ensure that at least one rule is contained in the match set. Further details on the process can be found in (Tomforde et al., 2009).

Layer 2: Off-line Learning

The existing set of classifiers and consequently the set of existing parameter sets has to be extended for situations where no classifier matches. This means, the system has to be able to autonomously learn parameter sets for unforeseen situations. Within our architecture, the Layer 2-component is responsible for this task. This component consists of three parts: an Observer, an EA and a simulator. The Observer is responsible for capturing the current situation description provided by the Observer on Layer 1. As the current usage of system components (in terms of CPU, RAM, etc.) has influence on the selection process of the LCS, the Observer is responsible for the scheduling of optimisation tasks.

The EA is responsible for evolving new classifiers. The algorithm is implemented as a standard Genetic Algorithm (cf. e. g. (Bäck and Schwefel, 1996) for details). This algorithm needs a possibility to analyse the performance of the current parameter set, which is done by using the standard network simulation tool *NS/2* (Web, 2009). The simulator needs a scenario and an implementation of the current protocol. The implementation is mandatory, but the configuration of the simulator depends on the observed situation as measured by the Observer on Layer 1. Therefore, a scenario is computed taking into account all observed attributes (e. g. for BitTorrent: number of peers, seeds, download and upload speeds, etc.)

4 RESEARCH ROADMAP

Within the previous Section the architecture of the system has been described. The system based on this has been realised and applied to a first protocol (BitTorrent - cf. (Cohen, 2003)). We demonstrated the potential of our system and validated the feasibility of our approach for a BitTorrent-based test scenario, leading to an increase in terms of the objective function (amount of downloaded data or download-time) of up to 20% (Tomforde et al., 2009). Further evaluation of this protocol is in progress. Additionally, we are working on demonstrating the applicability of our approach to other systems by replacing the SuOC (adapt and optimise mobile ad-hoc network protocols instead of a BitTorrent Client).

To completely achieve the goal as defined in Section 3.1 essential parts are still not investigated. The following part of this Section will emphasise the main focus of the future research based on this system. Therefore, we introduce the main research topics in accordance to the particular layers of our architecture.

4.1 Layer 0

The system aims at being generic in terms of controlling different protocols and protocol types. These protocols are situated at Layer 0 of our architecture as depicted in Fig. 1. To demonstrate the generic character of our approach we are going to apply the system to exemplary representatives of different protocol types.

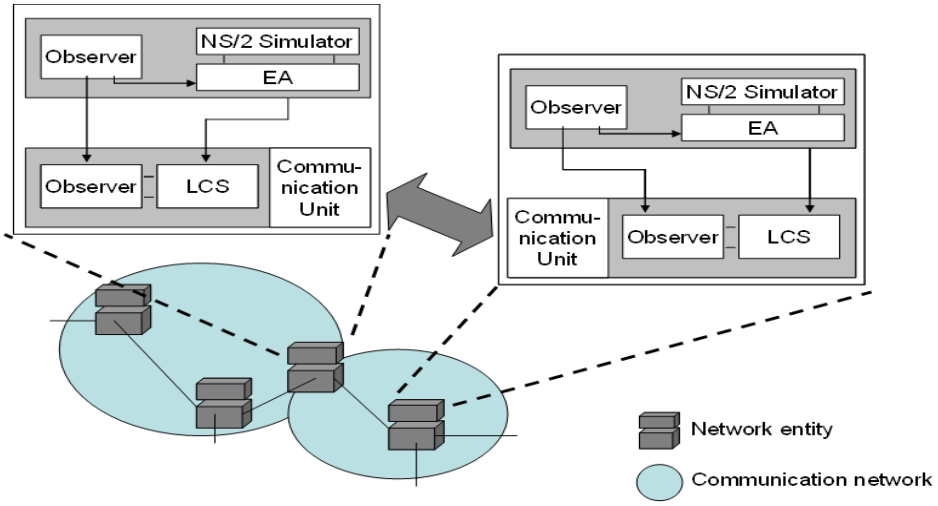


Figure 2: Collaborating network control systems.

Starting with BitTorrent as representative for Peer-to-Peer systems and the current application to mobile ad-hoc networks we will investigate the control of other protocol domains. Therefore, protocols for sensor networks are from interest as well as classical Internet protocols (like TCP/IP) and very specialised approaches like e. g. protocols for the communication in smart camera networks (Hoffmann et al., 2008).

In addition to the application of different protocol types, we aim at extending the control scope to *cross-layer optimisation* (Wang et al., 2005). This means e. g. for TCP/IP that the configuration of IP is selected depending on the current situation and of TCP depending on the configuration of IP.

4.2 Layer 1

The performance of the on-line adaptation mechanism at Layer 1 depends primarily on the applied learning technique. Due to this dependency we aim at validating the usage of our LCS by comparing it to other learning techniques. Therefore, existing techniques will be analysed based on the usage within our architecture and implemented if promising.

In addition to the increase of performance by analysing the learning component, the overall performance can be increased by allowing for collaboration. Neighboured entities should get the ability to collaborate with each other (see Fig. 2) in order to schedule Layer 2 tasks, exchange knowledge, and avoid redundant simulation-based learning.

4.3 Layer 2

As the off-line generation of new parameter sets is resource- and time-consuming, an improvement is necessary. The urgent target for this component is to speed up the rule creation process. The approach to solve this problem consists of two different strategies: a speed-up at start time and an approximation at runtime. At start time the system does not have any other rules than the standard parameter set of the protocol, which leads to the need of a fast mechanism to learn rules for a set of exemplary rules within the configuration space. These rules might have a lower quality than an optimised one, but they will be replaced with an optimised version during runtime.

The other aspect of the speed-up process at Layer 2 is, that nodes might not have sufficient resources for the optimisation task (e. g. sensor networks). Hence, research here will focus on approximating a reasonable parameter set (taking those parameter sets into account which are situated close to the situation description). This means, an intelligent inter- and extrapolation mechanism is needed in combination with a half-centralised solution (e. g. periodic updates from a service running on other nodes).

Another aspect of the Layer 2 optimisation is to use stand-by time (no active optimisation task) to optimise the coverage of the configuration space. This means some kind of active learning may be used to pro-actively generate parameter sets for situations where currently no adequate parameter set is known. Additionally, a collaboration mechanism will be helpful to schedule these active learning tasks for a set of neighboured entities.

5 CONCLUSIONS

This paper presented a system for the dynamic adaptation of network protocol parameters. The system monitors the situation at particular nodes and reacts on changes by adapting the communication protocol client. It is able to learn new control strategies and works on a self-organised basis. We explained our position that the presented system will be able to increase the performance of future communication networks without changing the whole technical background. Finally, we named the main research fields for our approach based on the introduced goal. Our system can also serve as a good testbed for the investigation of innate aspects of OC systems like trustworthiness or collaboration patterns.

REFERENCES

Bäck, T. and Schwefel, H.-P. (1996). Evolutionary computing: An overview. In *Proceedings of IEEE Conference of Evolutionary Computing*.

Cohen, B. (2003). Incentives Build Robustness in BitTorrent. In *Proceedings of the 1st Workshop on Economics of Peer-to-Peer Systems, Berkeley*.

Handley, M. (2006). Why the internet only just works. *BT Technology Journal*, 24(3):119–129.

Hoffmann, M., Wittke, M., Bernard, Y., Soleymani, R., and Hahner, J. (2008). Dmctrac: Distributed multi camera tracking. *Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on*, pages 1–10.

Jennings, B., van der Meer, S., Balasubramaniam, S., Botvich, D., Foghlu, M. O., Donnelly, W., and Strassner, J. (2007). Towards autonomic management of communications networks. *Communications Magazine, IEEE*, 45(10):112–121.

Kephart, J. O. and Chess, D. M. (2003). The Vision of Autonomic Computing. *IEEE Computer*, 36(1):41–50.

Khelil, A., Marron, P. J., Becker, C., and Rothermel, K. (2007). Hypergossiping: A generalized broadcast strategy for mobile ad hoc networks. *Ad Hoc Networks*, 5:531–546.

Montana, D. and Redi, J. (2005). Optimizing parameters of a mobile ad hoc network protocol with a genetic algorithm. In *GECCO '05: Proc. of the 2005 conference on Genetic and evolutionary computation*, pages 1993–1998, New York, NY, USA. ACM.

Prothmann, H., Rochner, F., Tomforde, S., Branke, J., Müller-Schloer, C., and Schmeck, H. (2008). Organic control of traffic lights. In *Proc. of the 5th Intern. Conference on Autonomic and Trusted Computing*.

Richter, U., Mnif, M., Branke, J., Müller-Schloer, C., and Schmeck, H. (2006). Towards a generic observer/controller architecture for Organic Computing. In Hochberger, C. and Liskowsky, R., editors, *INFORMATIK 2006 – Informatik für Menschen!*, volume P-93 of *GI-Edition – Lecture Notes in Informatics (LNI)*, pages 112–119. Köllen Verlag.

Schmeck, H. (2005). Organic Computing – A new vision for distributed embedded systems. In *Proceedings of the 8th IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC'05)*, pages 201–203.

Schmid, S., Sifalakis, M., and Hutchison, D. (2006). Towards autonomic networks. In *3rd Intern. Annual Conference on Autonomic Networking, Autonomic Communication Workshop (IFIP)*, Lecture Notes in Computer Science. Springer Verlag, Heidelberg.

Siekkinen, M., Goebel, V., Plagemann, T., Skevik, K.-A., Banfield, M., and Brusic, I. (2007). Beyond the future internet–requirements of autonomic networking architectures to address long term future networking challenges. *Future Trends of Distributed Computing Systems, IEEE International Workshop*, 0:89–98.

Sözer, E. M., Stojanovic, M., and Proakis, J. G. (2000). Initialization and routing optimization for ad-hoc underwater acoustic networks. In *Proc. of Opnetwork'00*.

Tomforde, S., Steffen, M., Hähner, J., and Müller-Schloer, C. (2009). Towards an organic network control system. *submitted for publication*.

Turgut, D., Daz, S., Elmasri, R., and Turgut, B. (2002). Optimizing clustering algorithm in mobile ad hoc networks using genetic algorithmic approach. In *Proc. of the IEEE Global Telecommunications Conference (GLOBECOM '02)*, pages 62 – 66.

Wang, J., Li, L., Low, S. H., and Doyle, J. C. (2005). Cross-layer optimization in TCP/IP networks. *IEEE/ACM Trans. Netw.*, 13(3):582–595.

Web (2009). The Network Simulator - NS/2. *http://www.isi.edu/nsnam/ns/*.

Wilson, S. W. (1995). Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149–175.

Ye, T., Harrison, D., Mo, B., Sikdar, B., Kaur, H. T., Kalyanaraman, S., Szymanski, B., and Vastola, K. (2001). Network Management and Control Using Collaborative On-line Simulation. In *Proceedings of IEEE ICC*, Helsinki, Finland. IEEE.

Ye, T. and Kalyanaraman, S. (2001). An adaptive random search algorithm for optimizing network protocol parameters. Technical report, Rensselaer Polytechnic Inst.

VISUAL SERVOING CONTROLLER FOR ROBOT MANIPULATORS

Jaime Cid and Fernando Reyes
Vicerrectoría de Investigación y Estudios de Posgrado, Grupo de Robótica
Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Electrónica
Apartado Postal 542, Puebla 72001, México
jcid@ece.buap.mx, freyes@ece.buap.mx

Keywords: Visual servoing, Control, Robot manipulator, Direct drive, Lyapunov function, Global asymptotic stability.

Abstract: This paper presents a new family of fixed-camera visual servoing for planar robot manipulators. The methodology is based on energy-shaping framework in order to derive regulators for position-image visual servoing. The control laws have been composed by the gradient of an artificial potential energy plus a nonlinear velocity feedback. For a static target we characterize the global closed loop attractor using the dynamic robot and vision model, and prove global asymptotic stability of position error for the control scheme, the so called position-based visual servoing. Inverse kinematics is used to obtain the angles of the desired joint, and those of the position joint from computed centroid. Experimental results on a two degrees of freedom direct drive manipulator are presented.

1 INTRODUCTION

The positioning problem of robot manipulators using visual information has been an area of research over the last 30 years. In recent years, attention to this subject has drastically grown. The visual information into feedback loop can solve many problems that limit applications of current robots: automatic driving, long range exploration, medical robotics, aerial robots, etc.

Visual servoing is referred to closed-loop position control for a robot end-effector using direct visual feedback (Hutchinson, 1996). This term was introduced by Hill and Park (Hill, 19179). It represents an attractive solution to position and motion control of autonomous robot manipulators evolving in unstructured environments.

On visual-servoing Weiss *et al.* (Weiss, 1987) and Wilson *et al.* (Wilson, 1996) have categorized two broad classes of vision-based robot control: position-based visual servoing, and image-based visual servoing. In the former, features are extracted from an image and used to estimate the position and orientation of the target with respect to the camera. Using these values, an error signal between the current and the desired position of the robot is defined in the joint space; while in the latter the error signal is defined directly in terms of image features to control the robot end-effector in order to move the image plane feature measurements to a set of desired locations. In both classes of methods, object feature points are mapped onto the camera image plane, and measurements of these points, for example a particularly useful class of image features are centroid used for robot control (Weiss, 1987; Wilson, 1996, Kelly, 1996).

In the configuration between camera and robot, a fixed-camera or a camera-in-hand can be fastened. Fixed-camera robotic systems are characterized in that a vision system fixed in the world coordinate frame, captures images of both the robot and its environment. The control objective of this approach is to move the robot end-effector in such a way that it reaches a desired target. In the camera-in-hand configuration, often called an eye-in-hand, generally a camera is mounted in the robot end-effector and provides visual information of the environment. In this configuration, the control objective is to move the robot end-effector in such a way that the projection of the static target be always at a desired location in the image given by the camera.

Since the first visual servoing systems were reported in the early 1980s the last few years have seen an increase in published research results. An excellent overview of the main issues in visual servo control of robot manipulators is given by (Corke, 1993). However, few rigorous results have been obtained incorporating the nonlinear robot dynamics. The first explicit solution of the problem formulated in this pa-

per was due to Miyazaki and Masutani in 1990, where a control scheme delivers bounded control actions belonging to the Transpose Jacobian-based family, philosophy first introduced by (Takegaki and Arimoto, 1981). Kelly addresses the visual servoing of planar robot manipulators under the fixed-camera configuration in (Reyes, 1998). Malis *et al* (1999) proposed a new approach to vision-based robot control, called 2-1/2-D visual servoing (Malis, 2005). The visual servoing problem is addressed by coupling the nonlinear control theory with a convenient representation of the visual information used by the robot in by (Conticelli, 2001).

(Park and Lee, 2003) present in a visual servoing control for a ball on a flat plate to track its desired trajectory. It has bee proposed in (Kellym 1996) a novel approach, they address the application of the velocity field control philosophy to visual servoing of the robot manipulator under a fixed-camera configuration. Schramm *et al* present a novel visual servoing approach, aimed at controlling the so-called extended-2D (E2D) coordinates of the points constituting a tracked target and provide simulation results (Reyes, 1997). Malis and Benhimane (2005) present a generic and flexible system for vision-based robot control. their system integrates visual tracking and visual servoing approaches in a unifying framework (Malis, 2003).

In this paper we address the positioning problem with fixed-camera configuration to position-based visual servoing of planar robot manipulators. Our main contribution is the development of a new family of position-based visual controllers supported by rigorous local asymptotic stability analysis, taking into account the full nonlinear robot dynamics, and the vision model. The objective concerning the control is defined in terms of joint coordinates which are deduced from visual information. In order to show the performance of the proposed family, two members have been experimentally tested on a two-degree-of-freedom direct drive vertical robot arm.

This paper is organized as follows. In Section 2, we present the robotic system model, the vision model and the formulation of the control problem, then the proposed visual controller is introduced and analyzed. Section 3 presents the experimental set-up. The experimental results are described in Section 4. Finally, we offer some conclusions in Section 5.

2 ROBOTIC SYSTEM MODEL

The robotic system considered in this paper is composed by a direct drive robot and a CCD-camera placed in the robot workspace in the fixed-camera configuration.

2.1 Robot Dynamics

The dynamic model of a robot manipulator plays an important role for simulation of motion, analysis of manipulator structures, and design of control algorithms. The dynamic equation of a n degrees of freedom robot in agreement with the Euler-Lagrange methodology (Spong, 1989), is given for

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + g(\mathbf{q}) = \tau \qquad (1)$$

where $\mathbf{q}$ is the $n \times 1$ vector of joint displacements, $\dot{\mathbf{q}}$ is the $n \times 1$ vector of joint velocities, τ is the $n \times 1$ vector of applied torques, $M(\mathbf{q})$ is the $n \times n$ symmetric positive definite manipulator inertia matrix, $C(\mathbf{q},\dot{\mathbf{q}})$ is the $n \times n$ matrix of centripetal and Coriolis torques, and $g(\mathbf{q})$ is the $n \times 1$ vector of gravitational torques.

It is assumed that the robot links are joined together with revolute joints. Although the equation of motion (1) is complex, it has several fundamental properties which can be exploited to facilitate the control system design. For the new control scheme, the following important property is used:

Property 1. The matrix $C(\mathbf{q},\dot{\mathbf{q}})$ and the time derivative $\dot{M}(\mathbf{q})$ of the inertia matrix both satisfy [12]:

$$\dot{\mathbf{q}}^T \left[\frac{1}{2}\dot{M}(\mathbf{q}) - C(\mathbf{q},\dot{\mathbf{q}}) \right] \dot{\mathbf{q}} = \mathbf{0} \;\; \forall\, \mathbf{q},\dot{\mathbf{q}} \in \mathbb{R}^n. \qquad (2)$$

2.1.1 Model of Direct Kinematic

Direct kinematics is a vectorial function that relate joint coordinates with Cartesian coordinates f: $\mathbb{R}^n \rightarrow \mathbb{R}^m$ where n is the number of degrees of freedom, and m represents the dimension of the Cartesian coordinate frame.

The position $x_R \in \mathbb{R}^3$ of the end-effector with respect to the robot coordinate frame in terms of the joint positions is given by: $x_R = f(\mathbf{q})$.

2.2 Vision Model

The goal of a machine vision system is to create a model of the real world from images. A machine vision system recovers useful information on a scene from its two-dimensional projections. Since images are two-dimensional projections of the three-dimensional world, this recovery requires the inversion of a many-to-one mapping (see Figure 1).

Let $\Sigma_R = \{R_1, R_2, R_3\}$ be a Cartesian frame attached to the robot base, where the axes R_1, R_2 and R_3

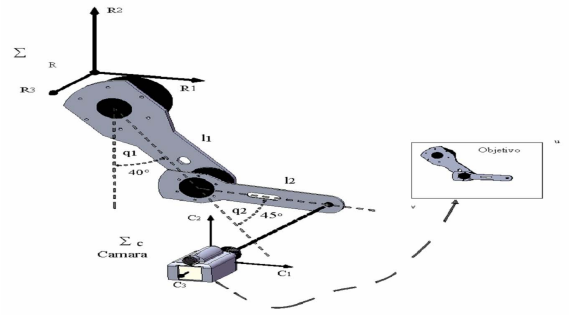


Figure 1: Fixed camera configuration.

represent the robot workspace. A CCD type camera has a $\Sigma_C = \{C_1, C_2, C_3\}$ Cartesian frame, whose origin is attached at the intersection of the optical axis with respect the geometric center of Σ_C. The description of a point in the camera frame is denoted by x_C. The position of the camera frame with respect to Σ_R is denoted by $o_C = [o_{C_1}, o_{C_2}, o_{C_3}]^T$.

The acquired scene is projected on to the CCD. To obtain the coordinates of the image at the CCD plane a perspective transformation is required. We consider that the camera has a perfect aligned optical system and free of optical aberrations, therefore the optical axis intersects at the geometric center of the CCD plane. Finally the image of the scene on the CCD is digitalized, transferred to the computer memory and displayed on the computer screen. We define a new two dimensional computer image coordinate frame $\Sigma_D = \{u,\ v\}$, whose origin is attached to the upper left corner of the computer screen. Therefore the vision system model is given by:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{\lambda}{\lambda + x_{C_3}} \begin{bmatrix} \alpha_u & 0 \\ 0 & -\alpha_v \end{bmatrix} \begin{bmatrix} x_{C_1} \\ x_{C_2} \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} x_{C_1} \\ x_{C_2} \\ x_{C_3} \end{bmatrix} = R^T(\theta)[x_R - o_R^c] \tag{4}$$

where $\alpha_u > 0$, $\alpha_v > 0$ are the scale factors in pixels/m, $\lambda > 0$ is the focal length of the camera and $\frac{\lambda}{\lambda + x_{C_3}} < 0$.

2.3 A New Position-based Visual Servoing Scheme for Fixed-camera Configuration

In this section, we present the stability analysis for the position-based visual servoing scheme. The robot task is specified in the image plane in terms of image feature values corresponding to the relative robot and object positions. It is assumed that the target resides in the plane $R_1 - R_2$, depicted in Figure 2. Let $[u_d \ \ v_d]^T$ be the desired image feature vector which is assumed to be constant on the computer image frame Σ_D. The desired joints q_d are estimated from inverse kinematic in function of $\begin{bmatrix} u_d & v_d \end{bmatrix}^T$.

The control problem in visual servoing for fixed-camera configuration consists in to designing a control law τ in such a way that the actual image feature $\begin{bmatrix} u & v \end{bmatrix}^T$ reaches the desired image feature $\begin{bmatrix} u_d & v_d \end{bmatrix}^T$ of the target. The image feature error is defined as $\begin{bmatrix} \tilde{u} & \tilde{v} \end{bmatrix}^T = \begin{bmatrix} u_d - u & v_d - v \end{bmatrix}^T$, therefore the control aim is to assure that $\lim_{t\to\infty} \begin{bmatrix} \tilde{q_1} & \tilde{q_2} \end{bmatrix}^T = \begin{bmatrix} q_{d_1} - q_1 & q_{d_2} - q_2 \end{bmatrix}^T \to 0$. The control problem is solvable if a joint motion q_d exists such that

$$\begin{bmatrix} u_d \\ v_d \end{bmatrix} = \begin{bmatrix} \frac{\alpha_u \lambda}{\lambda + x_{C_3}} & 0 \\ 0 & \frac{-\alpha_v \lambda}{\lambda + x_{C_3}} \end{bmatrix} R(\theta)^T \left(\begin{bmatrix} x_{R_1}(q_d) \\ x_{R_2}(q_d) \end{bmatrix} - \begin{bmatrix} o_{R_1}^c \\ o_{R_2}^c \end{bmatrix} \right). \tag{5}$$

In order to solve the visual servoing control problem, we present the next control scheme with gravity compensation:

$$\tau = \nabla \upsilon_a(k_p, \tilde{\mathbf{q}}) - f_v(k_v, \dot{\mathbf{q}}) + g(\mathbf{q}) \tag{6}$$

where $\tilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q} \in \mathbb{R}^n$ is the position error vector, $\mathbf{q}_d \in \mathbb{R}^n$ is the desired joint position vector $K_p \in \mathbb{R}^{n\times n}$ is the proportional gain which is a diagonal matrix, $K_v \in \mathbb{R}^{n\times n}$ is a positive definite matrix, also called derivative gain, $\nabla \upsilon_a(k_p, \tilde{\mathbf{q}})$ represents the artificial potential energy, being a positive definite function, and $f_v(k_v, \dot{\mathbf{q}})$ denotes the damping function, which is a dissipative function, that is, $\dot{q}^T f_v(k_v, \dot{\mathbf{q}}) > 0$.

Proposition. Consider the robot dynamic model (1) together with the control law (6), then the closed-loop system is global asymptotically stable, and the visual positioning aim

$\lim_{t\to\infty} \begin{bmatrix} \tilde{q}_1(t) & \tilde{q}_2(t) \end{bmatrix}^T = \mathbf{0} \in \mathbb{R}^2$ is achieved.

Proof: The closed-loop system equation obtained by combining the robot dynamic model (1) and control scheme (6) can be written as

$$\frac{d}{dt} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} -\dot{\mathbf{q}} \\ M(\mathbf{q})^{-1} \begin{bmatrix} \nabla \upsilon_a(\mathbf{k_p}, \tilde{\mathbf{q}}) - \mathbf{f_v}(\mathbf{k_v}, \dot{\mathbf{q}}) \\ -C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \end{bmatrix} \end{bmatrix} \tag{7}$$

which is an autonomous differential equation, and the origin of the state space is a equilibrium point. To carry out the stability analysis of equation (7), the following Lyapunov function candidate is proposed:

$$V(\widetilde{\mathbf{q}},\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}}^T M(\mathbf{q})\dot{\mathbf{q}} + \upsilon_a(\mathbf{k_p},\widetilde{\mathbf{q}}). \quad (8)$$

The first term of $V(\widetilde{\mathbf{q}},\dot{\mathbf{q}})$ is a positive definite function with respect to $\dot{\mathbf{q}}$ because $M(\mathbf{q})$ is a positive definite matrix. The second one of the Lyapunov function candidate (8), can be interpreted as a potential energy induced by the control law, and is also a positive definite function with respect to the position error $\widetilde{\mathbf{q}}$, because the term $\mathbf{k_p}$ is a positive definite matrix. Therefore, $V(\widetilde{\mathbf{q}},\dot{\mathbf{q}})$ is both a positive definite and radially unbounded function.

The time derivative of the Lyapunov function candidate (8) along the trajectories of the closed-loop equation (7), and after some algebra and considering property 1, can be written as

$$\begin{aligned}\dot{V}(\widetilde{\mathbf{q}},\dot{\mathbf{q}}) &= \dot{\mathbf{q}}^T M(\mathbf{q})\ddot{\mathbf{q}} - \frac{1}{2}\dot{\mathbf{q}}^T \dot{M}(\mathbf{q})\dot{\mathbf{q}} - \nabla\upsilon_a(\mathbf{k_p},\widetilde{\mathbf{q}})^T\dot{\mathbf{q}} \\ &= \dot{\mathbf{q}}^T\nabla\upsilon_{\mathbf{a}}(\mathbf{k_p},\widetilde{\mathbf{q}}) - \dot{\mathbf{q}}^T\mathbf{f_v}(\mathbf{k_v},\dot{\mathbf{q}}) - C(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} \\ &\quad + \frac{1}{2}\dot{\mathbf{q}}^T\dot{M}(\mathbf{q})\dot{\mathbf{q}} - \nabla\upsilon_a(\mathbf{k_p},\widetilde{\mathbf{q}})^T\dot{\mathbf{q}} \\ &= -\dot{\mathbf{q}}^T\,\mathbf{f_v}(\mathbf{k_v},\dot{\mathbf{q}}) \;\leq\; 0 \end{aligned} \quad (9)$$

which is a negative semidefinite function and therefore, it is possible to conclude stability in the equilibrium point. In order to prove local asymptotic stability, the autonomous nature of the closed-loop equation (7) is exploited to apply the LaSalle's invariance principle (Khalil, 2002) in the region Ω:

$$\Omega = \left\{ \begin{array}{c} \left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\\ \dot{\mathbf{q}}\end{array}\right] \in \mathbb{R}^{2n} :\; \dot{V}(\widetilde{\mathbf{q}},\dot{\mathbf{q}}) = 0 \\ \widetilde{\mathbf{q}} = 0 \in \mathbb{R}^n, \dot{\mathbf{q}} = 0 \in \mathbb{R}^n : \dot{V}(\widetilde{\mathbf{q}},\dot{\mathbf{q}}) = 0 \end{array} \right\} \quad (10)$$

since $\dot{V}(\widetilde{\mathbf{q}},\dot{\mathbf{q}}) \leq 0 \in \Omega$, $V(\widetilde{\mathbf{q}}(t),\dot{\mathbf{q}}(t))$ is a decreasing function of t. $V(\widetilde{\mathbf{q}},\dot{\mathbf{q}})$ is continuous on the compact set Ω, it is bounded from below on Ω. For example, it satisfies $0 \leq V(\widetilde{\mathbf{q}}(t),(t),\dot{\mathbf{q}}(t)) \leq V(\widetilde{\mathbf{q}}(0),\dot{\mathbf{q}}(0))$. Therefore, the trivial solution is the only solution of the closed-loop system (7) restricted to Ω. Consequently it is concluded that the origin of the state space is locally asymptotically stable.

2.4 Examples of Application

The purpose of this section is to exploit the methodology described above with the objective to derive new regulators.

We present control scheme with gravity compensation:

$$\tau = K_p \tanh\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right] - K_v \tanh\left[\begin{array}{c}\dot{\mathbf{q}}_1\\ \dot{\mathbf{q}}_2\end{array}\right] + g(q). \quad (11)$$

Proof: The closed-loop system equation obtained by combining the robot dynamic model (1) and control scheme (7) can be written as

$$\frac{d}{dt}\left[\begin{array}{c}\widetilde{\mathbf{q}}\\ \dot{\mathbf{q}}\end{array}\right] = \left[\begin{array}{c} -\dot{\mathbf{q}} \\ M(\mathbf{q})^{-1}\left[\begin{array}{c}\mathbf{k_p}\tanh(\widetilde{\mathbf{q}}) - \mathbf{K_v}\tanh(\dot{\mathbf{q}}) \\ -C(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\end{array}\right]\end{array}\right] \quad (12)$$

which is an autonomous differential equation, and the origin of the state space is a equilibrium point. To carry out the stability analysis of equation (12), the following Lyapunov function candidate is proposed:

$$\begin{aligned} V(\widetilde{\mathbf{q}},\dot{\mathbf{q}}) &= \frac{1}{2}\dot{\mathbf{q}}^T M(\mathbf{q})\dot{\mathbf{q}} + \sqrt{\ln\left(\cosh\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right]\right)}^T \\ &\quad K_p\sqrt{\ln\left(\cosh\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right]\right)} \end{aligned} \quad (13)$$

The first term of $V(\widetilde{\mathbf{q}},\dot{\mathbf{q}})$ is a positive definite function with respect to $\dot{\mathbf{q}}$ because $M(\mathbf{q})$ is a positive definite matrix. The second one of the Lyapunov function candidate (13), can be interpreted as a potential energy induced by the control law, and is also a positive definite function with respect to position error $\widetilde{\mathbf{q}}$, because the term $\mathbf{k_p}$ is a positive definite matrix. Therefore, $V(\widetilde{\mathbf{q}},\dot{\mathbf{q}})$ is both a positive definite and radially unbounded function.

The time derivative of Lyapunov function candidate (13) along the trajectories of the closed-loop equation (12) and after some algebra and considering property 1, can be written as:

$$\begin{aligned}\dot{V}(\widetilde{\mathbf{q}},\dot{\mathbf{q}}) &= \dot{\mathbf{q}}^T M(\mathbf{q})\ddot{\mathbf{q}} - \frac{1}{2}\dot{\mathbf{q}}^T\dot{M}(\mathbf{q})\dot{\mathbf{q}} - \dot{\mathbf{q}}^T K_p \tanh\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right] \\ &= \dot{\mathbf{q}}^T\tanh\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right] - \dot{\mathbf{q}}^T K_v \tanh\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right] \\ &\quad - C(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{q}}^T\dot{M}(q)\dot{\mathbf{q}} - \dot{\mathbf{q}}^T K_p\tanh\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right] \\ &= -\dot{\mathbf{q}}^T K_v \tanh\left[\begin{array}{c}\dot{\mathbf{q}}_1\\ \dot{\mathbf{q}}_2\end{array}\right] \leq 0 \end{aligned} \quad (14)$$

A second example from the proposed methodology:

$$\tau = K_p \arctan\left[\begin{array}{c}\widetilde{\mathbf{q}}_1\\ \widetilde{\mathbf{q}}_2\end{array}\right] - K_v \arctan\left[\begin{array}{c}\dot{q}_1\\ \dot{q}_2\end{array}\right] + g(\mathbf{q}). \quad (15)$$

3 EXPERIMENTAL SET-UP

An experimental system for research of robot control algorithms has been designed and built at the Universidad Autónoma de Puebla, México; it is a direct-drive robot of two degrees of freedom (see Figure 2). The experimental robot consists of two links made of 6061 aluminum actuated by brushless direct-drive servo actuators from Parker Compumotor in order to drive the joints without gear reduction. Advantages of this type of direct-drive actuator includes freedom from backlashes and significantly lower joint friction compared to actuators composed by gear drives. The motors used in the robot are listed in Table 1. The servos are operated in torque mode, so the motors act as a torque source and they accept an analog voltage as a reference of torque signal. Position information is obtained from incremental encoders located on the motors. The standard backwards difference algorithm applied to the joint positions measurements was used to generate the velocity signals. The manipulator workspace is a circle with a radius of 0.7 m. Besides position sensors and motor drivers, the robot also includes a motion control board, manufactured by Precision MicroDynamics Inc., which is used to obtain the joint positions. The control algorithm runs on a Pentium host computer.

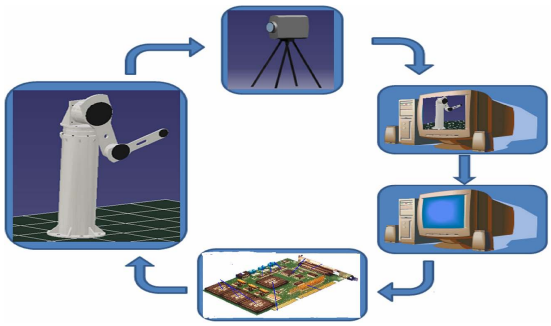


Figure 2: Experimental robot.

Table 1: Servo actuators of the experimental robot.

Link	Model	Torque	p/rev
Shoulder	DM1050A	50	1,024,000
Elbow	DM1004C	4	1,024,000

With reference to our direct-drive robot, only the gravitational torque is required to implement the new control scheme (6), which is available in (Malis, 2005):

$$g(\mathbf{q}) = \begin{bmatrix} 38.46\sin(q_1) + 1.82\sin(q_1+q_2) \\ 1.82\sin(q_1+q_2) \end{bmatrix} \text{ [Nm]}. \tag{16}$$

4 EXPERIMENTAL RESULTS

To support our theoretical developments, this Section presents experimental results of the proposed controllers on a planar robot for the fixed-camera configuration.

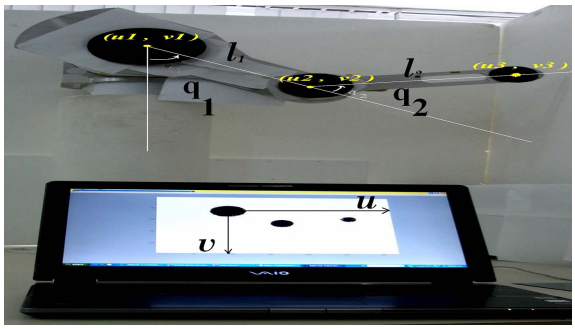


Figure 3: Robot manipulator and vision system.

Three black disks were mounted on the shoulder joint, elbow joint and end-effector, repestively. A big black disk for shoulder, a medium black disk on elbow, and a small one for the end-effector. The joint coordinates were estimated from predictable centroid using inverse kinematics as is shown in Figure (3):

$$l_1 = \sqrt{(u_2-u_1)^2+(v_2-v_1)^2} \tag{17}$$

$$l_2 = \sqrt{(u_3-u_2)^2+(v_3-v_2)^2} \tag{18}$$

and

$$q_2 = \arccos\left(\frac{(u_3-u_2)^2+(v_3-v_2)^2-l_1^2-l_2^2}{2l_1l_2}\right) \tag{19}$$

$$q_1 = \left(\frac{\pi}{2}\right) - atan\left(\frac{v_2-v_1}{u_2-u_1}\right) - atan\left(\frac{l_2\sin(q_2)}{l_1+l_2\cos(q_2)}\right) \tag{20}$$

where l_1, l_2 represent the link longitude respectively, u and v are visual information from equations (19) and (20) using in Figure 3.

The centroids of each disc were selected as object feature points. We select in all controllers the desired position in the image plane as $[u_d\ v_d]^T = [198\ 107]^T$ [pixels] and the following initial position $[u(0)\ v(0)]^T = [50\ 210]^T$ [pixels], this $q_1(0), q_2(0) = [0\ 0]^T$ and $\dot{\mathbf{q}}(0) = \mathbf{0}$ [degrees/sec]. The evaluated controllers have been written in C language. The sampling rate was executed at 2.5 ms., while the visual feedback loop was at 33 ms. The CCD camera was placed in front of the robot and its position with respect to the robot frame Σ_R was $O_c^R = [0_{c_1}^R, 0_{c_2}^R, 0_{c_3}^R]^T = [-0.5, -0.6, -1]^T$ meters, the rotation angle $\theta = 0$ degrees. We use MATLAB version 2007a, applying, the

SIMULINK module to carry out the imagine processing. The video signal from the CCD-camera has a resolution of 320x240 pixels in RGB format.

Figures 4 and 5 show the experimental results of the controller (11), the proportional and derivative gains were selected as $K_p = \text{diag}\{26.0,\ 1.8\}$ [N], $K_v = \text{diag}\{12.0,\ 1.2\}$ [Nm]. respectively and $\mathbf{u}_d = 198$ *y* $\mathbf{v}_d = 107$. The transient response with around 3 seconds is fast. The components of the feature position error tend asymptotically close to zero.

The experimental results for the controller (11) are shown in Figures (4) and (5). The transient response was around 3 seconds. The components of the feature position error tend asymptotically.

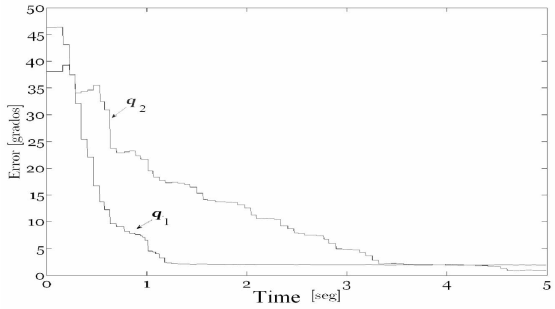


Figure 4: Error for controller (11).

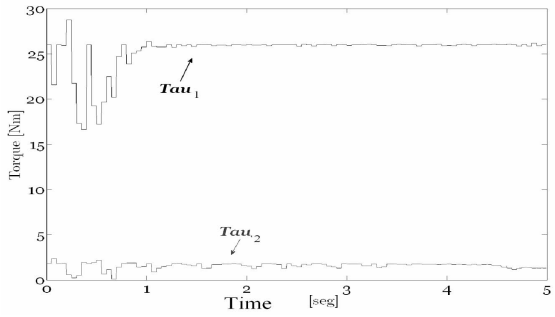


Figure 5: Torque for controller tanh.

Figures 6 and 7 show the experimental results of the controller (15). The proportional and derivative gains were selected as $K_p = \text{diag}\{17.3,\ 1.2\}$ [N], $K_v = \text{diag}\{6.6,\ 1.2\}$ [Nm], respectively, and $\mathbf{u}_d = 198$ *y* $\mathbf{v}_d = 107$. The transient response is fast by around 1 second. The components of the feature position error tend asymptotically to a neighborhood close to zero.

The experimental results for the controller (15) are shown in Figures 6 and 7. The transient response was around 1 second. The components of the feature position error tend asymptotically.

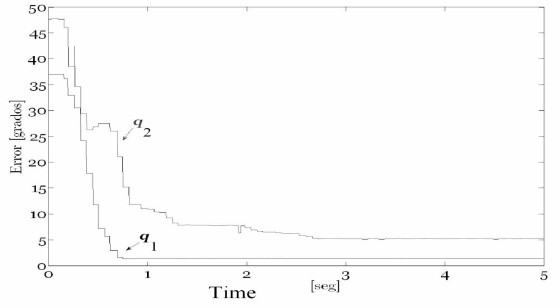


Figure 6: Error for controller (15).

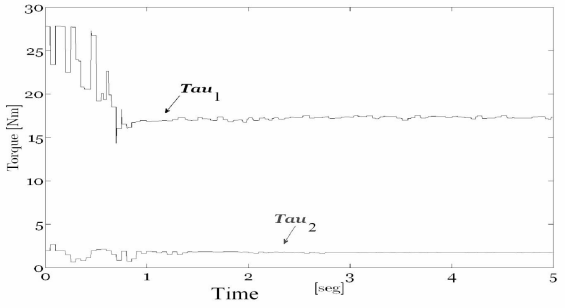


Figure 7: Torque for controller atan.

5 CONCLUSIONS

In this paper we have presented a new methodology to design position-based visual servoing for planar robots in fixed-camera configuration. It should be emphasized both that the nonlinear robot dynamics and the vision model have been included in the stability analysis. The class of controllers are energy-shaping based, and they are described by control laws composed of the gradient of an artificial potential energy plus a linear velocity feedback. Experimental results with a two degrees of freedom planar robot, using three feature points were presented to illustrate the performance of the control scheme.

REFERENCES

Hutchinson S., G. D. Hager and P. I. Corke, *A Tutorial on Visual Servo Control.* IEEE Trans. on Robotics and Automation, Vol. 12, No. 5, October 1996, pp. 651-670.

Hill J. and W. T. Park, *Real Time Control of a Robot with a Mobile Camera.* in Proc. 9th ISIR, Washington, D.C., Mar. 1979, pp. 233-246.

Weiss L. E., A. C. Sanderson, and C. P. Neuman, *Dynamic sensor-based control of robots with visual feedback.*

in IEEE Journal of Robot. Automat., vol. RA-3, pp. 404-417, Oct. 1987.

Wilson W. J., C. C. Williams, and Graham S. B. *Relative End-Effector Control Using Cartesian Position Based Visual Servoing.* IEEE Transactions on Robotics and Automation. vol. 12 No. 5, pp. 684-696. October 1996

Kelly R., P. Shirkey and M. W. Spong, *Fixed-Camera Visual Servo Control for Planar Robots.* IEEE International Conference on Robotics and Automation. Minneapolis, Minnesota, April 1996, pp. 2643-2649.

Corke P. I. *Visual Control of Robot Manipulators A review.* Visual Servoing, K. Hashimoto, Ed. Singapore: World Scientific, pp. 1-31, 1993.

Takegaki M. and S. Arimoto, *A New Feedback Method for Dynamic Control of Manipulators.* ASME J. Dyn. Syst. Meas. Control, Vol. 103, 1981, pp. 119-125.

Malis E. and S. Benhimane, *A Unified Approach to Visual Tracking and Servoing.* Robotics and Autonomous Systems, Vol. 52, Issue 1 , 31 July 2005, pp. 39-52.

Conticelli F. and B. Allotta, *Nonlinear Controllability and Stability Analysis of Adaptive Image-Based Systems.* IEEE Trans. on Robotics and Automation, Vol. 17, No. 2, 2001, pp. 208-214.

Park J. and Y.J. Lee, *Robust Visual Servoing for Motion Control of the Ball on a Plate.* Mechatronics, Vol. 13, Issue 7 , September 2003, pp. 723-738.

Malis E. and P. Rives, *Robustness of Image-based Visual Servoing with Respect to Depth Distribution Errors.* IEEE International Conference on Robotics and Automation. 2003, pp. 1056-1061.

Spong M. W. and M. Vidyasagar, *Robots Dynamics and Control.* John Wiley & Sons, 1989.

Reyes F. and R. Kelly, *Experimental Evaluation of Fixed-Camera Direct Visual Controllers on a Direct-Drive Robot.* IEEE International Conference on Robotics & Automation. Leuven, Belgium, May 1998, pp. 2327-2332.

Khalil, H. K. (2002). Nonlinear Systems. Prentice–Hall, Upper Saddle River, NJ.

Chen J., A. Behal, D. Dawson and Y. Fang, *2.5D Visual Servoing with a Fixed Camera.* American Control Conference. 2003, pp. 3442-3447.

Schramm F., G. Morel, A. Micaelli and A. Lottin, *Extended-2D Visual Servoing.* IEEE International Conference on Robotics and Automation. 2004, pp. 267-273.

Reyes F. and R. Kelly (1997). Experimental evaluation of identification schemes on a direct drive robot. *Robotica*, Cambridge University Press. Vol. 15, 563–571.

THE DEVELOPMENT OF A BRAKE BASED TORQUE VECTORING SYSTEM FOR A SPORT VEHICLE PERFORMANCE IMPROVEMENT

Leonidas Kakalis, Federico Cheli and Edoardo Sabbioni
Mechanical Department of Politecnico di Milano
Via la Masa 1, 20158 Milano, Italy
leonidas.kakalis@polimi.it, federico.cheli@polimi.it

Keywords: Vehicle dynamics, Passive differential, Semi-active differentials, Active differentials, Controlled brake system, Performance.

Abstract: In every driving condition powertrain and vehicle dynamics deeply influence each other. The main role of powertrain influence is played by the differential, which transmit the driving torque mainly with respect to wheel kinematics. Semi-active controlled versions of this device have been recently conceived to improve vehicle handling basing their function on the wheels kinematical conditions. On the other hand, active differentials allow to generate the most appropriate yaw moment controlling both the amount of transferred torque and its direction. The application presented in this the paper aims at enhancing the dynamic behavior of a rear-driven sport vehicle by creating the required yaw moment through brakes actuation and throttle control; the examined car is equipped with free differential, thus the proposed system does not require the introduction of additional devices. Performance measures relate to both open-loop and closed-loop driving demands, and include limit handling maneuvers.

1 INTRODUCTION

The conventional free differential is a mechanism that lets the driven wheels to assume different speeds while cornering with a uniform distribution of the driving torque on the two wheels of the same axle; this device shows its main limit when the adherence conditions of the two wheels are different: in this case a free differential is not able to transfer torque to the wheel with a higher adherence with the consequent result of a really poor traction of the vehicle. There are many examples of controlled differential systems in the literature (Pedrinelli, Cheli 2007) (Resta, Teuschl, Pedrinelli, Zorzutti 2005), (Zorzutti, Pedrinelli, Cheli, 2007). The vast majority employ a limited slip differential (LSD) similar to the passive gerodisc type where a friction clutch is employed effectively to provide a connection between the two drive shafts. The distinguishing feature of this type of LSD is that it will always transfer torque to the slower wheel. Such control systems thus have no control over the direction of torque transfer and are only able to modulate the applied magnitude.

The advent of the "overdriven" differential (Hancock, Williams, Gordon, Best, 2005), (Granzow, Gruhle, Spiess, Denzier, Baasch, Peter, 2007), (Leffler, 2007), however (Figure 1), makes it possible to control both the magnitude and direction of torque transfer. This allows the direction of the resulting yaw moment to be controlled and has led to the development of active yaw control systems (Tomari, Mori, Shibahata, 2005) which utilize controlled torque transfer. a powertrain equipped with an active differential system achieves an higher degree of flexibility: an active differential is designed to control both the locking torque (equivalent to the semi active one) and its direction; in this way it is possible to create a yaw moment regardless of the kinematical condition of the driven wheels, by transferring torque also from the slower one to the faster one. This flexibility produces a better compromise between traction and vehicle dynamics performance.

2 HANDLING TARGETS

The handling performance in steady state conditions of a high speed vehicle equipped with a semi-active differential is illustrated by the understeer curve reported in Figure 1.

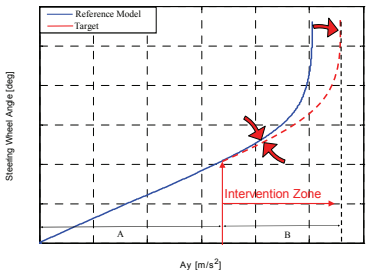


Figure 1: Reference understeer and target diagram.

Table 1: Selected test manoeuvres and performance targets.

Dynamic Conditions	Manoeuvre	Performance Index	Target
Steady State	Ramp Steer (ISO 4138:1996)	Understeer gradient	Reduce
		A_{ymax}	Increase
		Sideslip gradient	Increase
Power on	Ad-hoc	A_{ymax}, A_{xmax}	Increase
Mixed	Virtual race track	Time lap	Reduce
		Power dissipation	Reduce

The handling diagram of Figure 1 can be divided into two regions of interest: the linear region (A) where the response of the tires is still in the linear range and the non-linear region (B) where the tires gradually reach the frictional force saturation.

The generation of a yaw moment by means of active and semi-active differential or brake actuation can affect the shape of the understeer diagram both in region A and B; even if region B represents the zone of interest for a sport vehicle, where the main target is the highest lateral acceleration (red dotted line in Figure 1) achievable associated with vehicle stability. In Table 1 all the dynamic conditions among with their correspondent dynamic targets are summarized.

3 BRAKE TORQUE VECTORING SYSTEM (BTV)

The concept of BTV is based upon the generation of a yaw moment through independent brake actuation on the driven wheels. With respect to systems like Vehicle Dynamics Control (VDC), the main focus of BTV is the global enhancement of the vehicle performance; for this reason BTV acts also on the throttle valve to avoid the speed reduction associated with brake actuation. Even if the system is designed to increase lateral acceleration and promptness during transients, stability at limit is obviously included among the targets.

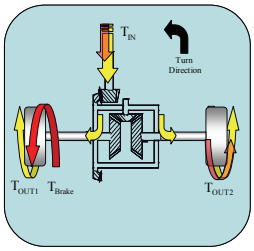


Figure 2: BTV general scheme. T_{IN}: input torque; T_{OUT1} and T_{OUT2}: resultant torque to each output shafts; T_{Brake}: braking torque.

Figure 2 represents a scheme of the BTV system intervention: assuming a steady state condition during a right turn of a rear driven car equipped with a free differential, a brake torque on the internal axle has been applied. As a consequence the external wheel must receive a torque equal to the braking torque applied on the internal one to keep the vehicle speed. The additional torque applied to the external wheel has been applied by accelerating the engine thus compensating energy dissipation produced by the brake actuation:

The asymmetric torque distribution on the rear axle can clearly affect the traction force balance and create a yaw moment mechanism. Compared to a passive or semi-active differential, this mechanism can be created independently from loading and adherence on the ground, imitating the function of an active differential. This implies that, during a turn, the system has the ability not only to transfer all the driving torque to the external wheel and maintain the internal one in free rolling condition but also to further amplify the yaw moment by creating a negative traction force on the internal wheel and increasing the driving torque on the external one.

4 CAR MODEL AND SIMULATION ENVIRONMENT

A 15 degrees of freedom model (IPG CarMaker®) of the examined sport car has been used to test and compare the performance of various control systems; the vehicle model has been integrated with the models of actuators and of the control logic implemented in Matlab/Simulink. The tires behavior has been described using MF-Tyre model version 2002 (Pacejka, 2003), taking into account combined slip effects. The model has been validated comparing its outputs to experimental data relevant to the passive vehicle equipped with a rear free differential.

In Figure 3 the understeer curve (steer angle vs. lateral acceleration) is shown for an ISO steering pad maneuver (ISO 4138:1996). The scaling is not reported in all figures in this paper because of confidentiality agreements.

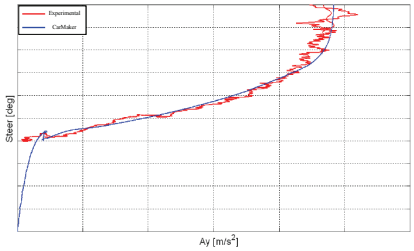


Figure 3: Steering pad constant radius. Understeering curve numerical vs. experimental.

Such a relative validation has allowed achieve a better confidence in the presented numerical results.

Due to the significance of a proper clutch stiction and slip phenomena modelling, the powertrain model has been developed using a mathematical approach appropriate for this kind of analysis (Cheli, Pedrinelli, Zorzutti, 2007).

5 CONTROL LOGIC ARCHITECTURE

In paragraph 3 the target of this project has been pointed out as the maintenance of the stability at limit and, above all, the vehicle performance improvement in regard to the same car equipped with a semi-active differential or an active differential.

The control logic is not based on modern control theory (LQR, etc.), but the simpler way of a feed-forward to guaranty a quick response and PID controllers to better adjust the overall algorithm output is chosen.

First of all the algorithm foresees that the car state has to be detected (Kakalis, 2009): the system, then, applies dedicated sub-algorithms, one for steady-state/step steer/power on and one for power off (Kakalis, 2009) which results the desired brake torque.

5.1 Steady State

As said in the previous sections, the resultant yaw moment should not lead to an oversteering condition. Therefore, the control system must work only when it can guarantee a sensible gain in vehicle performance. Because of that the feed-forward part is constituted of a 3D map whose values correspond to the maximum oversteering moment tolerable by the car in various adherence levels. The applied yaw moment should follow certain rules. At low speed and lateral acceleration the gain in terms of understeering gradient is narrow so that the driver shouldn't perceive a major handling improvement. On the other hand, at high lateral acceleration the gain in maximum lateral acceleration should be hugely influenced by the logic intervention.

Based on the 3D map, BTV is capable of generating a high asymmetry distribution (braking inner wheel) of the rear longitudinal forces due to the simultaneous action on brakes and throttle. However, such an extreme torque vectoring can generate an uncomfortable feeling (tank steering), so that a standard lateral torque distribution (LTD) was imposed on both the models (Figure 5) where, as limit case, the internal wheel is kept in free rolling condition.

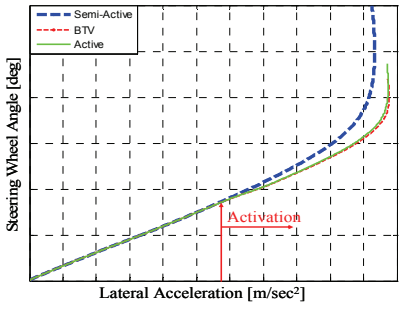


Figure 4: Understeer curve.

Figure 4 presents an example of an understeer curve for a fixed velocity of 100km/h, comparing the response of the same car equipped with semi-active differential, active differential and BTV. It can be easily noted that beyond 6 m/s^2 (activation

threshold) the understeer gradient is reduced for both BTV and active differential; both the systems have produced a better exploitation of the frictional forces thus allowing to reach higher lateral acceleration with respect to the semi-active differential (+5%). The examination of Figure 6 suggests that this improvement is obtained with an increase of the sideslip angle.

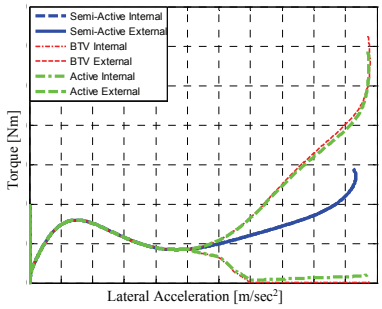


Figure 5: Lateral torque distribution.

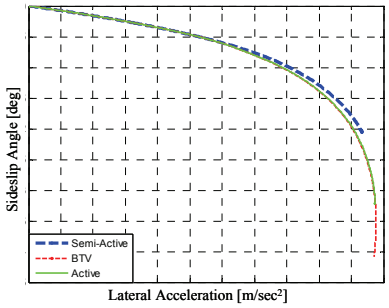


Figure 6: Sideslip angle.

5.2 U-turn (Power on)

The maintenance of the longitudinal acceleration thresholds under medium and high lateral acceleration imposes the need to combine two fundamental arguments: stability in the limit area and optimal traction.

As examined before the control logic of both BTV and active differential in steady-state cornering, generates a strongly asymmetric torque distribution (0–100%). However, if a simultaneous longitudinal acceleration is required by the driver, the external tire can't guarantee alone all the traction force and the lateral one without saturating and generating oversteer. Because of that, the transferred torque to the external wheel must be limited by changing the distribution ratio, i.e. the internal wheel should be progressively accelerated. This action will reduce the inwards moment whose amplitude is directly governed by the longitudinal dynamic state of the inner wheel. The acceleration of the inner wheel causes the longitudinal slip boost and thus the longitudinal force increase. It's important to not exceed the longitudinal slip peak (normally around 12% and 14%) to avoid the wheel spinning and a huge engine rpm increase. In case of BTV the progressive reduction of the torque distribution is necessary and it is particularly complicated also because an excessive braking action would dissipate a lot of engine power that could be used to accelerate the vehicle.

The optimization of the longitudinal slip is based on a PID controller. The error signal is given by the difference between the actual longitudinal slip and the optimal one (12%-14%).

As far as BTV is concerned, the controller directly commands the braking torque applied on the inner wheel while in the active differential regulates the outer clutch.

In order to compare the performance of the three models under power on conditions, an ad-hoc maneuver was designed (Figure 7), consisting of two parts: in the first one the vehicle enters a curve and progressively reaches steady-state conditions (Steady State phase) achieving maximum performance (maximum velocity and lateral acceleration). In this part of the maneuver both BTV and the active differential impose a 0-100% LTD ratio. It has to be underlined that, in order to extract meaningful conclusions, the driver model forces the three vehicles to follow the same trajectory.

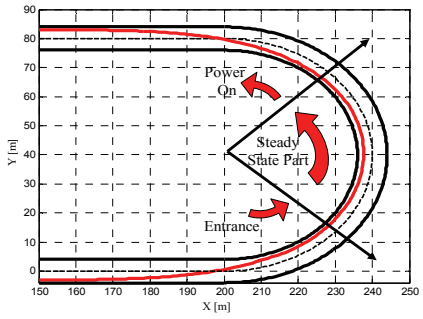


Figure 7: U-turn (Radius = 40m).

The second part (Power On phase) begins when the driver accelerates (full throttle) and exits the curve following the defined trajectory. During the steady state phase BTV and active differential clearly show their superiority in respect to the semi-active model by describing the fixed trajectory with a higher velocity (+2%).

As far as the transient phase (power on) is concerned, BTV accelerates several meters before the semi-active model and the active one. Any

attempt, for both the vehicle equipped with semi-active and active differential, to accelerate before would cause an oversteering response and exit from the track.

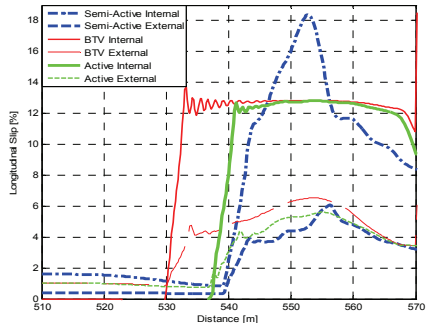


Figure 8: U-turn. Longitudinal slip on internal and external wheel vs. distance.

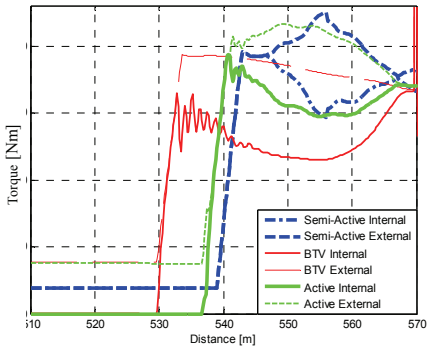


Figure 9: U-turn. Net torque on the real left and right semi-axle vs. distance.

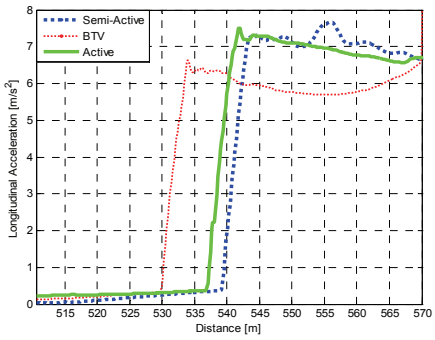


Figure 10: U-turn. Longitudinal acceleration during the exit phase vs. distance.

The need to follow the reference longitudinal slip (Figure 8) would produce an excessive drive torque transfer to the outer wheel (Figure 9) causing its saturation. The BTV yaw moment generation mechanism is instead more flexible since the torque on the inner wheel can be controlled without the need of transferring the same torque to the external one. Such property makes it possible to initiate the power on phase much earlier. Although both the semi-active and the active differential lead the vehicle to accelerate several meters after BTV, both the systems allow a better exploitation of the remaining longitudinal adhesion and achieve a higher longitudinal acceleration (Figure 10).

Judging by the distance history of the longitudinal velocity (Figure 11), BTV slightly improves the performance of the active differential and it presents a considerable advantage over the semi-active.

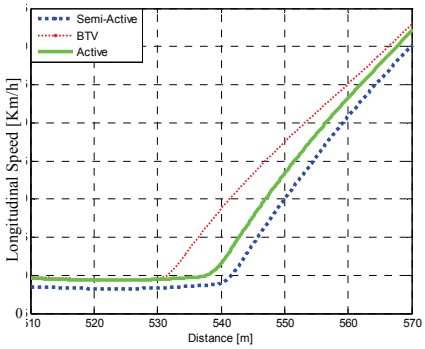


Figure 11: U-turn. Longitudinal speed vs. distance.

5.3 Virtual Race Track

As a last test, the performance offered by the three control systems was tested comparing their performance on an entire race track.

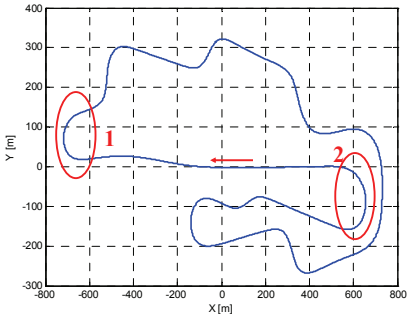


Figure 12: Selected race track.

The choice to validate the performance for all the three systems on the virtual track of Figure 12, showed the need to increase the robustness of their logic in order to extract more meaningful results. Such a test implies the fact that all models should have a common state-recognizing switch governed by the same principles and then the same power-off strategy in order to eliminate great trajectory

variations. The simulations were carried out in CarMaker^{TM} environment; the virtual driver of IPG^{TM} was chosen to perform the simulations with a driving style very close to the one of a real driver.
Active differential and BTV have been actuated by the same control logic previously presented for steady-state curve and power on transient; this implies that in steady-state the internal wheel does not transmit any traction force to the ground. Once power-on conditions is recognized, the optimization of the internal's wheel longitudinal slip will take place.

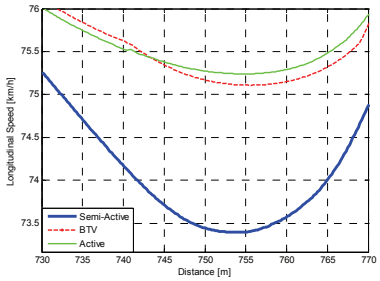


Figure 13: Longitudinal speed vs. distance in turn 1.

Considering a total lap time of approximately 140 s, BTV and active differential allow a reduction of 1.9% and 2.0% respectively when compared to the performance produced by the vehicle equipped with the semi-active differential. The time difference between the three systems can be explained by analyzing the dynamic performance in different circuit sections. By observing at the first turn speed profile (Figure 13) BTV and active differential achieve a longitudinal velocity 2.5% higher with respect to vehicle equipped with the semi-active differential.

6 TEMPERATURE ANALYSIS

6.1 Brake Temperature Estimation

Increased power dissipation produced by repeated brake actuations, might pose concerns around their temperature and efficiency; it is therefore required to estimate the expected temperature increase in the brake system to complete the feasibility analysis of the proposed concept. It has to be underlined that the authors feel to provide only a short description of the developed thermal model because its complexity and the assumptions taken into consideration would require a more detailed analysis which can be found in (Sabbioni, Cheli, 2008) and (Limpert, 1999).

The thermal model takes into consideration the heat transfer due to conduction between:

- the rotor and the braking pad;
- the braking pad and the caliper;
- rotor and disc's hub;
- disc's hub and wheel carrier;
- and to forced convection between:
- caliper, rotor and braking pad and the air;
- disc's hub and wheel carrier with the air;

The validation of the numerical model was carried out by using ten consecutive laps test results recorded on a race track using as a test vehicle the reference model equipped with the semi-active differential. The temperature was measured through a temperature sensor positioned in the braking pad.

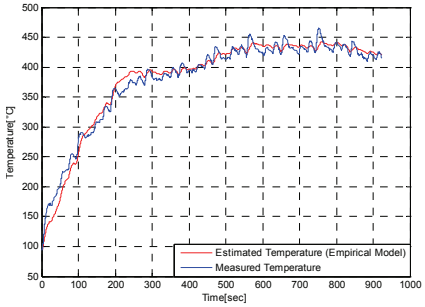


Figure 14: Comparison between measured and estimated braking pad temperatures.

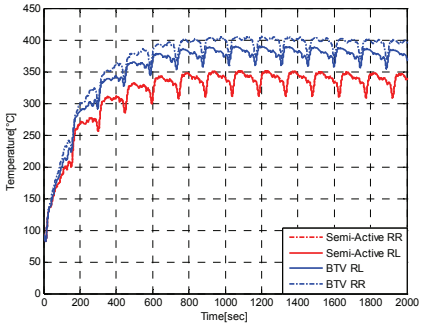


Figure 15: Brake pad temperature estimation.

Figure 14 shows the comparison between the temperature measured on the braking pad and the one obtained by the numerical brake model.
The brake model was fed with the data obtained through the simulation on the test track; this procedure allowed to estimate the discs temperature time history and thus evaluate the increased thermal load associated with the BTV logic. Figure 15 collects the results obtained from the control systems: the brake temperature gradually increases with time and reaches a mean operating temperature after about 5-6 laps. In terms of temperature, the

dissipated power differences presented in Figure 15 correspond to a disc's temperature rise of approximately 50 ^{o}C for the rear left brake and 75 ^{o}C for the rear right one. This temperature difference between the two models may be considered limited and tolerable since telemetry data on the real car indicated operating temperatures above 400 ^{o}C (Figure 15).

7 CONCLUSIONS

This paper presented the feasibility study of a system designed for the improvement of the handling characteristics of a sport vehicle based on the yaw moment control. The proposed system, named BTV, generates an asymmetric distribution of the longitudinal forces on the driving axle through an independent actuation of the brakes and a control of the throttle valve. As far as handling performance is concerned, BTV showed its superiority with respect to the semi-active differential and allows to get the same improvement provided by an active differential under several operating conditions. Besides this, BTV presents an important advantage related to its implementation on a real vehicle which would not require any additional electronic or mechanical component. On the other side, active differential still appears superior as far as the mechanism of generation of the yaw moment is concerned: BTV produces a torque difference by dissipating the energy supplied to one of the wheels in the form of heat, while the active differential simply attempts to reapportion the torque that is supplied to the wheels. The mechanism by which this is achieved - the friction clutch - still leads to some energy loss, but this is generally much lower than the energy dissipated in the brakes. The low energy consumption of the active differential gives it the potential to apply yaw moment control throughout the operating range of the vehicle.

The increased thermal solicitation of the brake system was also examined through a thermal model of brakes; according to the model results the expected increase of the temperature of the discs after a series of laps on a race track will not compromise the brake efficiency.

Obviously remains still in discussion the problem of the adherence level identification. This difficult task can be handled through the definition and the implementation of a self-governing recognizing algorithm which, based on the observation of the on board measured sizes, can replace the manual control regulation made by the driver which now is the implemented solution on the reference vehicle. A major step towards the adherence recognition can be considered the new generation of Cyber Tires, (Pasterkamp, Pacejka, 1997), (Mancosu and others, 2008).

REFERENCES

Milliken, W.F., Milliken, D.L., 1995, SAE. *Race Car Vehicle Dynamics*.

Pacejka, H.B, 2002. *Tyre and vehicle dynamics*. Oxford: Butterworth-Heinemann.

Pedrinelli, M., Cheli, F., 2007. *Vehicle Dynamics control system actuating an active differential*. SAE paper.

Pedrinelli, M., Zorzutti, A., Cheli, F., 2007. *Development of a control strategy for a semi-active differential for a high performance vehicle*. SAE paper.

Cheli, F., Pedrinelli, M., Zorzutti, A., 2007. *Integrated vehicle and driveline modelling*. SAE paper.

Hancock, M.J., Williams, R.A., Gordon, T.J., Best, M.C., 2005. *A comparison of braking and differential control of road vehicle yaw-sideslip dynamics*. Proc. of the Institution of Mechanical Engineers.

Zanten, v.A., Erhardt, R., Pfaff, G., 1995. *VDC, The Vehicle Dynamics Control System of Bosch.* SAE paper.

Granzow, C., Gruhle, W.D., Spiess, M., Denzier R., Baasch., D., Peter, R., 2007. *Driving Precision by Torque Vectoring–the new ZF axle drive*. Proc. of the European All-Wheel Drive Congress, Graz.

Leffler, H., 2007. *xDrive and Vehicle Dynamics Control of the new BMW X5*. Proc. of the European All-Wheel Drive Congress, Graz.

Shibahata, Y., Shimada, K., Tomari, T., 1993. *Improvement of vehicle maneuverability by Direct Yaw Moment Control*. Vehicle System Dynamics, 22.

Limpert R., 1999. Brake Design and Safety-*2nd Edition*. SAE

Sabbioni E., Cheli F., 2008. *A Numerical Model for Predicting Thermal Loads in Passenger Car Brake Disks*. Proc. VSDIA, Budapest.

Kakalis, L., *A brake based torque vectoring system for sport vehicle improvement*. PhD thesis.

Pasterkamp, W.R., Pacejka, H.B., 1997. *The tire as a sensor to estimate friction*. Vehicle System Dynamics, 27, 409-422.

Mancosu, F. and others, 2008. *Method and system for determining a tyre load during the running of a vehicle*. Patent No.PT0283 WO P0.

ISO 4138:1996. *Passenger cars – Steady state circular driving behavior – Open loop test procedure*.

CoP3D: CONTEXT-AWARE OVERLAY TREE FOR CONTENT-BASED CONTROL SYSTEMS*

Mauro Caporuscio
Dipartimento di Elettronica de Informazione, Politecnico di Milano, Italy
caporuscio@elet.polimi.it

Alfredo Navarra
Dipartimento di Matematica e Informatica, Università degli Studi di Perugia, Italy
navarra@dmi.unipg.it

Keywords: Content-based publish/Subscribe system, Sensor networks, Distributed computing, Overlay network.

Abstract: Publish/Subscribe systems are nowadays largely accepted for coordinating applications over wide-area networks, whereas they still suffer when applied to mobile environments. In fact, the typical tree overlay network implementation of the event service, which provides scalability in the context of wide-area networks, fails when a run-time reconfiguration is needed. This might happen when dealing with the dynamic and mobility characteristics inherent to the pervasive environments. In this proposal, we address such an issue by envisioning a context-aware approach in order to build and maintain the dispatching tree overlay with respect to the context sensed through the environment. The resulting distributed control system optimizes on the energy consumption aspect, which is the key cost measure for a mobile environment such as sensor networks.

1 INTRODUCTION

Continuous improvement of wireless network technologies and the miniaturization of electronic equipments are de facto enabling the exploitation of wireless sensor networks as distributed control systems for monitoring data within physical environments. In particular, a sensor network consists of a number of sensor nodes randomly disseminated within the environment. This requires for protocols and algorithms that let nodes to self-organize and cooperate with each other.

In this context, content-based publish/subscribe system is considered. Indeed, it is a well suited communication infrastructure for dealing with distributed applications over wireless networks. In fact, the loose coupling characteristic of publish/subscribe together with content-based routing protocol, where the message routing is determined by the interests of the receiver rather than by the explicit destination address, allow for dealing with the dynamism of such environments (Carzaniga and Wolf, 2001).

As depicted in Figure 1, a content-based publish/subscribe system is composed of two main entities: clients and servers. *Servers* are interconnected in a distributed network, referred to as *event service*, and provide clients with *access points* offering an extended publish/subscribe interface. *Clients* are of two kinds: (*i*) *publishers* use the access points to *publish* events and (*ii*) *subscribers* use the access points to *subscribe* for events of interest by supplying a predicate, called *subscription*. A *subscription* is applied to the content of events and it allows *subscribers* to select the events they are interested in. The *event service* is responsible for selecting and delivering events of interest to subscribers via the access points. The idea of a publish/subscribe system is quite mature, and a fair number and variety of publish/subscribe systems have been proposed, including research prototypes (Carzaniga et al., 2001; Cugola et al., 2001;

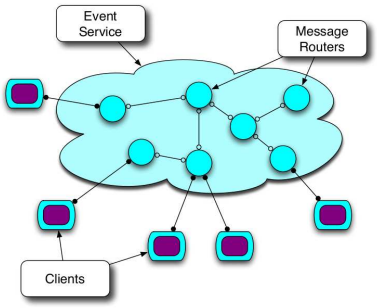


Figure 1: Tree overlay implementation of the event service.

*This research has been partially funded by the European Commission, Programme IDEAS-ERC, Project 227077-SMScom.

Pietzuch and Bacon., 2002; Meier and Cahill, 2002), commercial products (TIBCO Inc., Palo Alto, CA, 1996), and attempts of standardization (Object Management Group (OMG), 2004; Sun Microsystems, Inc., Mountain View, California, 1999).

The typical implementation of the event service is a tree overlay network (as shown in Figure 1), which provides scalability in the context of wide-area networks (e.g., Siena (Carzaniga et al., 2001) and Elvin (Segall and Arnold., 1997)). In fact, having a unique path between any two nodes simplifies both the matching algorithm and the routing scheme, and avoids network flooding. Even though such an overlay topology has been successfully exploited in ad-hoc environments (see e.g., multicast tree (Royer and Perkins, 2000)) there are issues concerning its application to publish/subscribe systems in the context of mobile environments. In fact, the high degree of dynamics inherent to sensor networks requires to continuously reconfigure the event service and, consequently, to reconfigure data structures and routing tables. Such structures, distributed through the event service, are created by considering the interest expressed by the clients and evolve at runtime when subscriptions are added/removed/modified. This makes the reconfiguration process expensive (in terms of both message exchanges and computational operations) and hard to accomplish.

Many works have been published so far concerning publish/subscribe systems in mobile environments, such as (Huang and Garcia-Molina, 2004; Fiege et al., 2003). However, most of them focus on the clients mobility while the event service remains stable (Caporuscio et al., 2003). On the other hand, in (Huang and Garcia-Molina, 2003) and (Mottola et al., 2008), the authors concentrate on the event service topology reconfiguration in the field of ad-hoc networks.

This paper deals with publish/subscribe systems over sensor network by proposing a context-aware technique to build and maintain the dispatching tree overlay with respect to the context sensed through the environment. Our proposal is to exploit the Connectionless Probabilistic (CoP) protocol (McCann et al., 2005), originally designed for sensor networks, in the context of publish/subscribe systems. The CoP protocol assumes battery powered mobile devices that interact with each other in order to establish a virtual infrastructure for routing purposes. Namely, a virtual grid infrastructure is built by considering the physical position of the devices. Further, the protocol takes care of the energy consumption by benefiting from mobility and network dynamics.

We envision an extension of the CoP protocol in order to build a rooted tree on top of the virtual grid. This would realize a new protocol, called CoP3D, where the tree is used to accomplish publish/subscribe transmissions (i.e., the overlay network), whereas the underlying grid maintains its original aim. The main property of the proposed CoP3D protocol is that it builds the overlay network taking into account context information sensed through the environment. Namely, the devices representing the tree nodes will be selected according to their physical location, available energy, computational resources and expected mobility rate.

The paper is organized as follows. Section 2 describes the CoP protocol and the virtual grid construction. Section 3 presents CoP3D and describes how the tree overlay is built with respect to the sensed context. Section 4 points out interesting peculiarities of the considered context that must be carefully addressed. Section 5 discusses future work.

2 CoP: THE VIRTUAL GRID INFRASTRUCTURE

In this section, we outline the CoP protocol (McCann et al., 2005), which aims to manage communication for sensor networks. We have chosen this protocol since it efficiently performs in mobile ad-hoc environments and well fits our purposes.

We assume the devices can move over some given area A. From now on we refer to devices also by nodes. As it will be clear soon, our approach can be adapted to any kind of desired shape for A. It may be suitable for any building like offices, airports, companies and so forth where the network has to be deployed. For the sake of clarity, A is assumed to be simply a square area of side d. At the border of A there can be some fixed infrastructure, if required, that supports the survivability of the network. In the context of publish/subscribe systems, whenever a publisher wants to provide a service on the network, it has to inform the network about its publication. In the wired context this message was routed along a fixed tree responsible of the matching among publishers and subscribers. Since in our wireless and mobile context we cannot have such an infrastructure, the basic idea is to build a virtual one along which messages are routed.

Virtual infrastructure means that active nodes have to take care somehow of its feasibility and maintenance. The only thing that each node needs to know in order to participate in the network, by means of the CoP protocol, is its own physical position with respect to the area A. This can be achieved by means of either devices powered by GPS systems or some in-

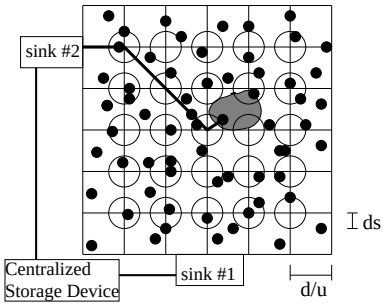


Figure 2: The area of interest covered by a virtual grid infrastructure of grid unit $\frac{d}{u}$. Nodes inside each circle of radius ds compete to represent the corresponding grid node.

stalled beacon external nodes that inform about relative positions (the mentioned infrastructure at the border of A), or other positioning techniques (see for instance, (Capkun et al., 2006; Caruso et al., 2005)). According to the expected population inside A, CoP builds a virtual infrastructure. Namely, a square grid G covering A is considered with grid unit d/u (see Figure 2), where u is determined according to the density of the nodes. From the "balls into bins" theory (see (Raab and Steger, 1998; McCann et al., 2005)), by choosing an appropriate distortion parameter ds, according to the density of the nodes in the area of interest, we can compute our desired value u in such a way that for each grid node there will be some device inside the circle of radius ds centered on it with high probability. We can choose for instance $u = \frac{d}{3ds}$. Note that u can change according to the established density of the network over the time. Its current value can be part of the beacon message sent by the external border nodes.[2]

Whenever a device in A starts its interaction with the network or just moves from its current position, it is aware about its location with respect to A and hence to G. This allows for determining whether it can be elected as representative of a grid node. For this step of the CoP virtual grid construction, we refer to standard leader election strategies suitably adopted in wireless environments like ad-hoc and sensors networks (see (Malpani et al., 2000)).

In doing so, at the expenses of some communication needed for the evaluation of a suitable value for u and the local leader elections, CoP builds a virtual grid G that can be used in order to correctly route desired communications. G covers all the given area A. One of the most important properties of such a construction is that it does not depend on the current nodes acting as leaders but only on the density of the nodes. The CoP protocol was originally applied in the field of sensor networks where sensed data needed to be route outside A to a fixed sink. Hence, the routing was performed in a multi-hop fashion over the grid. The main steps that must be performed by a sensor in order to participate to the CoP protocol can be summarized as follows.

1. It discovers its position according to the used technology, the area covered by the network and the size of the grid needed for the routing
2. If the received value for u is zero then it just waits
3. According to the previous information it evaluates its position with respect to the grid and decides whether it can represent any grid node or not. If not, it just assumes there is someone representing the closest grid node to its position and whenever needs to send a message it delegates such a node for the correct routing. If yes, it checks whether there is someone else playing or not that role
4. If it represents a grid node then it has to take care of the associated traffic that has to be route
5. One hop of the communication is made over the grid in such a way that the transmission can be received inside the whole circular area associated to the target grid nodes. This is due to the fact that a transmitting node does not know the exact position of its neighbors but it only assumes that someone is inside the circle of radius ds surrounding the target grid node
6. Whenever a device changes its position, i.e., either enters or exits a circular area, it has to change its role accordingly. If it was responsible for the routing, it has to take care to send the needed information to at least one of the current candidate to become its successor
7. If a device is running out of energy, it has to become a passive node even though still continues to receive its desired communications until it can.

Starting from this routing protocol and infrastructure, next section shows how this can be modified and suitably adapted in the case of mobile publish/subscriber systems.

3 CoP3D: THE TREE OVERLAY NETWORK

In this section, we describe our proposal concerning the management of a mobile publish/subscriber system. In particular, we exploit the CoP virtual grid construction in order to obtain a tree overlay implementing the event service. The main differences with re-

[2] $u = 0$ means there are not enough devices in the area to build the network.

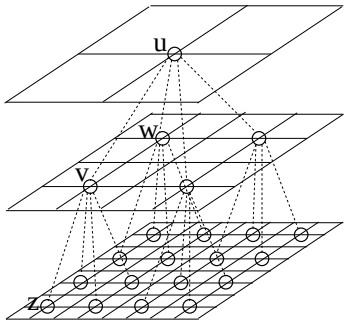


Figure 3: The tree construction based on the virtual grid infrastructure.

spect to a typical structured environment can be summarized as follows:

(a) *Energy Efficiency*: mobile devices have scarce power capacity and they can be required to stay alive for long periods without any support

(b) *Scalability*: it is desirable that any solution to manage such type of network would easily scale according to the number of devices and the corresponding area. Due to the mobility feature, manual deployment of devices is just not feasible

(c) *Fault-tolerance*: even though wired networks are also usually designed to cope with fault-tolerance issues, in a mobile environment faults are much more frequent. A device can easily disappear from the network due to either its movement or its low power level

(d) *Absence of the Infrastructure*: Due to their movements or to some handoff strategy, a given device is not always linked to the same set of neighbors at different time. This implies that assumptions about the topology of the network cannot be done unless they are straightforward from the area of interest (for instance networks along a street cannot differ too much from a line). The only fixed infrastructure that can be assumed can reside at the border of the area of interest

As shown in Figure 3, virtual grid nodes become the leaves of the tree overlay. Over the basic virtual grid infrastructure previously described, we build several levels of grids. What in the typical publish/subscribe system was the routing tree now is a 4-ary tree rooted at the center of the grid. Dividing the grid into four sub-grids we iteratively discover other 4 centers (one for each subgrid) to which the previous one is virtually connected. We iterate such a process $2\log\frac{1}{u}$ times in order to obtain a full coverage of the grid by means of a 4-ary tree of logarithmic height with respect to the number of grid nodes (see Figure 3). Starting from the bottom layer, once the basic virtual grid is built, all the nodes also know their location with respect to the overlay tree and hence, each one can play its corresponding role. By referring to Figure 3, for instance, node *v* knows that in the overlay tree it plays the role of parent for node *z*.

As it was for the virtual grid construction, many devices may cleverly compete in a leader election in order to become nodes of the tree overlay, hence responsible for routing publish/subscribe events and subscriptions. In particular, such an election is performed by considering context information. These include: (*i*) available energy and physical location to be compliant with the CoP protocol (see Section 2), (*ii*) mobility rate since the more a node is stable the more a node is qualified to play the router role, and (*iii*) computational resources since powerful nodes can better accomplish routing tasks.

In this way, the devices representing the virtual grid nodes of the 4-ary tree are responsible for the routing of the messages to realize the desired publish/subscribe system. Such nodes will be clearly more loaded than others and hence they will spend more energy. In order to distribute the energy consumption, changes to the grid structure can be done by means of shifting procedures or changing the grid unit *u*. It can also be implicitly obtained by means of a certain mobility ratio. Whenever a device changes its position, in fact, it has to evaluate if its role is also changing according to the distortion parameter *ds*. When a node moves out from the virtual circular area of radius *ds* defined around each grid node, it cannot be anymore representative of such a grid node hence it has to delegate someone else for doing its job. This means that mobility can play a central role in this routing process since it implies a more uniform distribution of the energy consumption. Once the tree overlay network is built, the event service can act as in the case of the typical structured environment. This allows us to apply such an approach to any publish/subscribe system that relies on a tree overlay network.

Concerning the properties outlined at the beginning of this section, we want to point out how our approach is suitable with respect to them. For (a) it is worth to note that the only resource of the devices is their energy. Moreover, both the virtual grid infrastructure and the tree overlay are heavily dependent on the devices survivability. On the other hand, this is the only resource that we can rely on in order to avoid the installation of typical structured environments. The tree overlay network construction is quite lightweight since it requires only some basic calculations. Furthermore the routing is well spread among all the devices and it is made in a multi-hop fashion. It takes

into account both the maximum distance covered by means of a transmission (hence the maximum energy spent) and the time needed to deliver a message (that in the worst case requires only $4\log\frac{1}{u}$ steps). As last remark, our strategy makes use of the mobility rate as a mean to save energy since roles change accordingly.

Concerning (b) and (c), our approach is easily scalable. It is in its nature to adapt the network coverage based on the density of the devices with respect to the area that must be served. It also suitably copes with fault-tolerance issues. In fact, devices might continuously appear (disappear) in (from) the network without breaking virtual connections up to a certain threshold. Once reached such a threshold, communications among the devices become unfeasible.

The last property (d) is easily solved by means of the virtual grid infrastructure, hence avoiding the installation of both structured environments and more powerful devices.

4 FURTHER KEY-POINTS

We now focus the attention on three main properties that must be carefully addressed in our approach.

(*i*) As it was for structured event services, it is evident that starting from the root and descending until the leaves, nodes are differently loaded in terms of routing messages. As already outlined, one mean to avoid this unbalanced energy consumption might depend on the mobility rate or the shifting procedure. On the other hand it is worth noting that our construction can be easily modified in the case we prefer to not deploy the root at the center of the grid since we may have information about the distribution of the devices. The previous discussion is in fact based on the assumption of a uniform distribution of the devices inside *A*. Consequently, it is easy to understand that the area of interest can be subdivided into several areas according to the expected density, hence applying our methodology to construct different subtree overlays.

(*ii*) Another very important issue that must be addressed before practically apply our method concerns the maximum distance that devices' transmissions can cover. With our construction, in fact, not only the root is the most loaded node in terms of communications but also it is required to perform the most distant transmissions overall the network. In order to avoid such a situation and to better balance the energy consumption among the nodes of the network, we may think about a reversed structure. This implies that the more a node is far from the root (with respect to the tree overlay), the more must be its transmission distance. In Figure 4 it is shown how the tree overlay

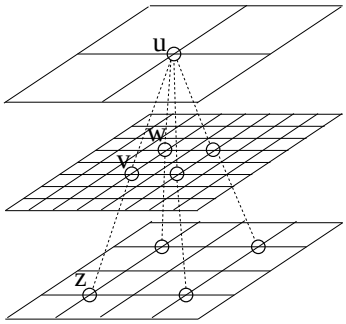


Figure 4: The tree construction based on the reversed virtual grid infrastructure. The aim is to better balance energy consumption among the devices.

may become. In any case, when a device belongs to a layer in which its power cannot cope with the required transmission distance, it does not mean that it cannot perform the required communications. In the header of the messages can be added information in such a way that a long transmission, in terms of covered distance, can be emulated by multi-hop transmissions of shorter range. This recalls what was used in the field of ATM networks by the concepts of Virtual Channel and Virtual Path communications (see e.g., (Flammini and Navarra, 2009)). It is not in the aim of this paper to go into details of such discussion but we want only point out that the underlying virtual grid structure at each layer can be used for this purpose.

(*iii*) Finally, CoP was based on another assumption for which some communications can be wasted due to the absence of a sensor representative of some virtual grid node. According to the parameter *ds*, such an event can happen rarely but still there is the possibility. Since in our context the tree overlay must be somehow guaranteed until it is feasible, we need some strategy to cope with this possibility. Due to the underlying virtual grid structure, in fact, the transmissions directed to a missing node can be easily rerouted to some other grid path. For this issue we can refer to (Mostarda and Navarra, 2008) where CoP was modified in order to address security issues. Clearly, at the expenses of a higher energy consumption, such situations can be addressed.

5 FUTURE WORK

In this paper, we have envisioned a new approach to achieve content-based publish/subscribe control systems in the context of sensor networks. In particular, we have proposed a context-aware technique to build and maintain the tree overlay network for communication purposes. Specifically, our approach builds on

the CoP protocol and takes into account context information sensed through the environment. In fact, the devices representing the tree nodes are chosen by considering their physical location, available energy, computational resources and expected mobility rate. Once the tree overlay network is built, the publish/subscribe system can act as in the case of the typical structured environment. This allows to apply such an approach to any publish/subscribe system that relies on a tree overlay network. It is of main interest to study both the theoretical feasibility and the effectiveness of our proposal. Moreover, our methodology might be also applied to other distributed systems rather than just the publish/subscribe ones (e.g., peer-to-peer) and to build other topology overlay networks.

REFERENCES

Capkun, S., Cagalj, M., and Srivastava, M. (2006). Secure localization with hidden and mobile base stations. In *Proc. of the* 25^{th} *Annual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM)*. IEEE Computer Society.

Caporuscio, M., Carzaniga, A., and Wolf, A. L. (2003). Design and evaluation of a support service for mobile, wireless publish/subscribe applications. *IEEE Transactions on Software Engineering*, 29(12):1059–1071.

Caruso, A., Chessa, S., De, S., and Urpi, A. (2005). GPS free coordinate assignment and routing in wireless sensor networks. In *Proc. of the* 24^{th} *Annual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM)*, pages 150–160. IEEE Computer Society.

Carzaniga, A., Rosenblum, D. S., and Wolf, A. L. (2001). Design and evaluation of a wide-area event notification service. *ACM Transactions on Computer Systems*, 19(3):332–383.

Carzaniga, A. and Wolf, A. L. (2001). Content-based networking: A new communication infrastructure. In *NSF Workshop on an Infrastructure for Mobile and Wireless Systems*, number 2538 in Lecture Notes in Computer Science, pages 59–68, Scottsdale, Arizona. Springer-Verlag.

Cugola, G., Nitto, E. D., and Fuggetta, A. (2001). The JEDI event-based infrastructure and its application to the development of the OPSS WFMS. *IEEE Transactions on Software Engineering*, 27(9):827–850.

Fiege, L., Grtner, F. C., Kasten, O., and Zeidler, A. (2003). Supporting mobility in content-based publish/subscribe middleware. In *Lecture Notes in Computer Science*, volume 2672, pages 103 – 122.

Flammini, M. and Navarra, A. (2009). Layouts for mobility management in wireless atm networks. *Discrete Applied Mathematics (DAM)*, 157(1):98–111.

Huang, Y. and Garcia-Molina, H. (2003). Publish/subscribe tree construction in wireless ad-hoc networks. In *Proc. of the* 4^{th} *Int. Conf. on Mobile Data Management (MDM)*, pages 122–140.

Huang, Y. and Garcia-Molina, H. (2004). Publish/subscribe in a mobile environment. *Wireless Networks*, 10(6):643–652.

Malpani, N., Welch, J. L., and Vaidya, N. H. (2000). Leader election algorithms for mobile ad hoc networks. In *Proc. of ACM Joint Work. on Foundations of Mobile Computing (DIALM-POMC)*.

McCann, J. A., Navarra, A., and Papadopoulos, A. A. (2005). Connectionless Probabilistic (CoP) routing: an efficient protocol for Mobile Wireless Ad-Hoc Sensor Networks. In *Proc. of the* 24^{th} *IEEE Int. Performance Computing and Communications Conf. (IPCCC)*, pages 73–77.

Meier, R. and Cahill, V. (2002). Steam: Event-based middleware for wireless ad hoc networks. In *Proc. of the* 1^{st} *Int. Work. on Distributed Event-Based Systems (DEBS)*, Vienna, Austria.

Mostarda, L. and Navarra, A. (2008). Distributed idss for enhancing security in mobile wireless sensor networks. *Int. Journal of Distributed Sensor Networks (IJDSN)*, 4(2):83–109.

Mottola, L., Cugola, G., and Picco, G. P. (2008). A self-reparing tree topology enabling content-based routing in mobile ad hoc networks. *Transaction on Mobile Computing*, 7(8):946–960.

Object Management Group (OMG) (2004). Notification service version 1.1.

Pietzuch, P. R. and Bacon., J. (2002). Hermes: A distributed event-based middleware architecture. In *Proc. of the* 1^{st} *Int. Work. on Distributed Event-Based Systems (DEBS)*, pages 611–618, Vienna, Austria.

Raab, M. and Steger, A. (1998). "Balls into bins" - a simple and tight analysis. In *Proc. of the* 2^{nd} *Int. Work. on Randomization and Approximation Techniques in Computer Science (RANDOM)*, pages 159–170.

Royer, E. and Perkins, C. (2000). Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing. IETF, Internet Draft: draft-ietf-manet-maodv-00.txt.

Segall, B. and Arnold., D. (1997). Elvin has left the building: A publish/subscribe notification service with quenching. In *Proc. of the Australian UNIX User Group*, pages 243–255, Brisbane, Australia.

Sun Microsystems, Inc., Mountain View, California (1999). Java message service.

TIBCO Inc., Palo Alto, CA (1996). TIBR+: a WAN router for global data distribution.

OPTIMAL CONTROL OF HAZARDOUS MATERIALS TRAFFIC FLOW

The Case of Transport through a Critical Infrastructure

Chiara Bersani, Riccardo Minciardi, Michela Robba, Roberto Sacile and Angela Maria Tomasoni
DIST, Department of Communication, Computer and System Sciences, Italy
Michela.robba@unige.it

Keywords: Optimal control, Hazardous materials, Transport, Traffic.

Abstract: In this work, a preliminary study as regards the possibility to define optimal control strategies for the hazmat (hazardous material) traffic flowing towards one critical road infrastructure (e.g. as in the case study a tunnel) at the macroscopic level is introduced. Specifically, the simplified model that is studied is related to part of a highway, on which the hazmat traffic can flow from one entrance. The control variables are represented by the number of vehicles that are allowed to enter the highway during a specific time interval, while the state variables are the queue of vehicles before the entrance, the number of vehicles in the various tracts of the highway, and the number of vehicles that enter the tunnel. The objective function to be minimized is characterized by three main terms: the queue, the hazard over the road, and the hazard related to the tunnel.

1 INTRODUCTION

Hazardous materials cover a wide range of products (explosives, gases, flammable liquids and solids, radioactive materials, hazardous wastes, etc. (Verter and Kara, 2008)). Transportation of these materials (that is, in general, multi-modal: road, pipelines, railway, ship) is a relevant problem to be considered because of the significant amount of material that flows among roads, territory and infrastructures (Bersani et al., 2008). Defining strategies for hazardous materials (hazmat) transportation management is a complex task because it is necessary to take into account different objectives (minimize risks, satisfy goods demand transportation), different decision makers (fleet managers, local authorities, infrastructures managers), and different approaches (mainly based on the different spatial-temporal scales to be considered: strategic planning, tactical planning, operational management).

In the literature of hazardous materials transportation on road, there are few, thought important and relevant, works on this subject (for example: Berman et al., 2007; Verter and Kara, 2007; Kara and Verter, 2004; Sadjadi, 2007; Bell, 2009; Bell and Cassir, 2002, Bersani et al., 2008a; Serafini, 2006; Beroggi and Wallace, 1994). The majority of these works is based on optimization models for planning and design purposes. The preliminary approach presented in this work is instead based on real time operational management (like the work presented by Bersani et al., 2008b) with specific reference to the case of critical infrastructures.

The transportation of hazardous materials (hazmat) on road has important consequences in the overall traffic management (Minciardi et al., 2008). This fact is more evident when a vehicle requires to move towards a critical road infrastructure, such as a tunnel or a bridge. The control of traffic networks has been the subject of a great amount of literature from different viewpoints. The main articles related to the case of a tunnel are reported in (Minciardi et al., 2008). The aim of this preliminary study regards the possibility to define optimal control strategies for the hazmat traffic flowing towards one critical road infrastructure (e.g. as in the case study a tunnel).

A given number of vehicles transporting hazardous material has to use a highway and to reach one critical infrastructure (e.g. a tunnel). They can stop in a park before the highway entrance and start their travel according to the exigencies of a decision maker that can be identified as the tunnel manager. The park may be taken into account as an

inventory in which the state of the system is represented by the vehicles that are present at a specific time instant. The flow dynamics of hazardous material vehicles on the highway has also to be modelled. In particular, the problem is defined at a macroscopic level, in which the state and the control variables correspond to the number of vehicles, for which the integrity condition may be relaxed, in order to obtain a continuous-variable decision problem. The control variables are represented by the number of vehicles that are allowed to enter the highway during a specific time interval, while the state variables are the queue of vehicles before the entrance, the number of vehicles in the various tracts of the highway, and the number of vehicles that enters the tunnel. The objective function to be minimized is characterized by three main terms: the queue, the hazard over the road, and the hazard related to the tunnel.

The resulting optimal control problem is linear quadratic with non-negativity constraints over the state and control variables. A receding horizon control scheme is used to derive the solution and to allow the model to be suitable in real time decision frameworks. An optimization package (Lingo 9.0, www.lindosystems.com) is used to solve the problem at each step.

In fact, the explicit form of the optimal control law of a given linear, discrete-time, time-invariant process subject to a quadratic cost criterion is well known in the unconstrained case, while, even for simple constraints, solution is hard to achieve. In (Castelein and Johnson, 1989), the authors use the controllable block companion transformation and derive sufficient conditions on the weighting matrices of the cost criterion to ensure that the closed-loop response of the original process with the standard, unconstrained optimal feedback law will be nonnegative. Bertsimas and Brown (2007) assess that the celebrated success of dynamic programming

for optimizing quadratic cost functions over linear systems is limited by its inability to tractably deal with even simple constraints, and present an alternative approach based on results from robust optimization to solve the stochastic linear-quadratic control (SLQC) problem.

For this reason, interesting developments of this work will be devoted to the definition of methodologies to find efficient solutions for the optimal control strategies.

In the next subsections, the system model is described in detail. Then, the decision problem is formalized. Finally, results and conclusion are drawn.

2 THE SYSTEM MODEL

Figure 1 shows the schematic representation of the decision framework: the highway directed towards one critical infrastructure is modelled as a line divided in highway tracts. As a simplification, two highway tracts have been considered.

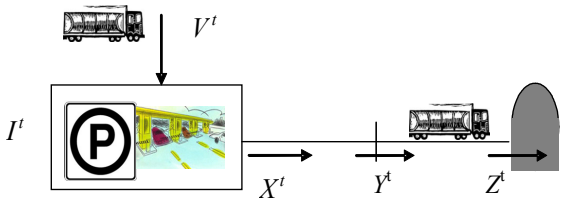


Figure 1: The considered system.

The physical inputs of the whole system are the quantities V^t, i.e., the (known) number of vehicles entering the park near the highway entrance in time interval $(t, t+1)$, $t = 0,\ldots,T\text{-}1$. The control variables correspond to the number of vehicles that enter the highway X^t in a specific time interval $(t, t+1)$, while the state variables correspond to the number of vehicles in the inventory/queue, I^t, the number of vehicles per tract of the highway (N_1^t, N_2^t), and the number of vehicles going out from tract 1 and entering the tunnel (Y^t, Z^t).

Two different kinds of state equations have to be introduced, regarding, respectively, the queue in the park at the highway entrance, and the highway tracts. Moreover, the hazard has been formalized as a function of the state and control variables.

2.1 The Queue State Equation

The state equation is:

$$I^{t+1} = (I^t + V^t - X^t) \qquad t=0,\ldots, T\text{-}1 \qquad (1)$$

where:

- I^t is the number of vehicles stored, at time instant t, in the park near the entrance, i.e., the inventory of the entrance park area, in time interval $(t, t+1)$;
- X^t is the number of vehicles that enter the highway in time interval $(t, t+1)$, from the entrance park area;
- V^t is the (known) number of vehicles that enters the entrance park in time interval $(t, t+1)$.

2.2 The Highway Tract State Equations

These state equations describe the evolution over time of a state variable that represents the number of hazmat vehicles (per unit length) present in a specific tract of the highway. The speed of these vehicles is related to the overall vehicle density over the considered tract. It is assumed that the vehicle flow can be represented through an average speed, which is common to hazmat and non-hazmat vehicles. In agreement with the literature dealing with traffic models, it is assumed that the (average) vehicle speed is never so high to allow the complete covering of a highway tract within a single time interval (of course, this may be also seen as a constraint over the space discretization of the highway). The equations are given by

$$N_1^{t+1} = N_1^t + \frac{X^t}{L_1} - \frac{Y^t}{L_1} \qquad t=0,\ldots,T\text{-}1 \qquad (2)$$

$$N_2^{t+1} = N_2^t - \frac{Z^t}{L_2} + \frac{Y^t}{L_2} \qquad t=0,\ldots,T\text{-}1 \qquad (3)$$

with

$$Y^t = N_1^t vel_1^t \Delta t \qquad t=0,\ldots,T\text{-}1 \qquad (4)$$

$$Z^t = N_2^t vel_2^t \Delta t \qquad t=0,\ldots,T\text{-}1 \qquad (5)$$

where:

- N_1^t, N_2^t are the number of (hazmat) vehicles per unit length that is present in the highway road in tracts 1 and 2, in time instant t;
- L_1, L_2 are the tracts lengths;
- Δt is the time interval length;
- vel_1^t, vel_2^t are the (average) velocities in the tracts in time interval (t, t+1), which is assumed to be imposed by the ordinary traffic (i.e., non hazmat), assuming that the hazmat vehicle flow is only a negligible part of the overall traffic flow;
- Y^t is the number of vehicles that passes from tract 1 to tract 2 in time interval (t, t+1);
- Z^t is the number of vehicles that reaches the tunnel in time interval (t, t+1).

2.3 Hazard Assessment

The hazard of accidents depends on different structural and environmental parameters that may vary for each time interval and for each highway tract, and on the number of vehicles (Fabiano et al., 2002; Fabiano et al., 2005). In this work, the hazard HAZ^t is simply represented as a time-varying a-dimensional parameter $\eta_{HAZ}{}^t$ multiplied by the number of vehicles in the specific tract. That is,

$$HAZ^t = \eta_{HAZ}{}_1^t N_1^t L_1 + \eta_{HAZ}{}_2^t N_2^t L_1 + \eta_{HAZ}{}_3^t Z^t$$
$$t=0,\ldots,T\text{-}1 \qquad (6)$$

3 THE DECISION PROBLEM

The objective function has to take into account the number of vehicles in the park entrance, the number of vehicles per unit length in each tract of the highway, and the number of vehicles that enter the tunnel. In particular the following terms have to be minimized:

- the number of vehicles waiting in the park entrance;
- the number of vehicles per unit length for tract 1, N_1^t;
- the number of vehicles per unit length for tract 2, N_2^t;
- the difference between the number of vehicles per unit length in tract 1 and tract 2, $N_1^t - N_2^t$;
- the number of vehicles that enter the tunnel.

Thus, the objective function can be expressed as

$$\min \sum_{t=0}^{T-1} \left(I^t\right)^2 + \alpha\left(N_1^t\right)^2 + \beta\left(N_2^t\right)^2 + \\ + \gamma\left(N_1^t - N_2^t\right)^2 + \delta\left(Z^t\right)^2 \qquad (7)$$

where:

- N_1^t, N_2^t are the number of (hazmat) vehicles per unit length that is present in the highway road in tracts 1 and 2, in time instant t;
- I^t is the number of vehicles stored, at time instant t, in the park near the entrance, i.e., the inventory of the entrance park area, in time interval (t, t+1);
- Z^t is the number of vehicles that reaches the tunnel in time interval (t, t+1);
- α, β, γ, δ are specific weighting factors.

4 THE STATEMENT OF THE OPTIMAL CONTROL PROBLEM

The optimal control problem reported in equations (1)-(7) can be expressed in the following form

$$\min_{u_t} \quad \sum_{t=0}^{T-1} \underline{x}_t{}^T Q_t \underline{x}_t \tag{8}$$

where $\underline{x}_t$ is the space vector and Q_t a matrix of time dependent parameters. Specifically,

$$\underline{x}_t = \begin{bmatrix} I^t \\ N_1^t \\ N_2^t \end{bmatrix} \qquad t=0,\ldots,T\text{-}1 \tag{9}$$

$$Q_t = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha+\gamma & -\gamma \\ 0 & -\gamma & \beta+\gamma+\delta vel_t^2 \Delta t^2 \end{bmatrix} \quad t=0,\ldots,T\text{-}1 \tag{10}$$

s.t.

$$\underline{x}_{t+1} = A_t \underline{x}_t + \underline{b} u_t + \underline{d}_t \qquad t=0,\ldots,T\text{-}1 \tag{11}$$

$$u_t \geq 0 \qquad t=0,\ldots,T\text{-}1 \tag{12}$$

$$\underline{x}_t \geq 0 \qquad t=0,\ldots,T\text{-}1 \tag{13}$$

where $u_t = X_t$ are the control variables, A_t is a matrix of time dependent parameters, $\underline{b}$ a vector of parameters, and $\underline{d}_t$ a vector of time dependent parameters.

$$A_t = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1-\frac{vel_1^t \Delta t}{L_1} & 0 \\ 0 & \frac{vel_1^t \Delta t}{L_2} & 1-\frac{vel_2^t \Delta t}{L_2} \end{bmatrix} \quad t=0,\ldots,T\text{-}1 \tag{14}$$

$$\underline{b}_t = \begin{bmatrix} -1 \\ \frac{1}{L_1} \\ 0 \end{bmatrix} \qquad t=0,\ldots,T\text{-}1 \tag{15}$$

$$\underline{d}_t = \begin{bmatrix} V^t \\ 0 \\ 0 \end{bmatrix} \qquad t=0,\ldots,T\text{-}1 \tag{16}$$

The optimal control problem expressed by equations (8)-(16) is a linear-quadratic one, with non negativity constraints over the state and control variables.

5 RESULTS

The space-time discretization of equations (2)-(3) has been chosen in order to avoid instability of the traffic flow (i.e., in the time interval, the vehicles are not allowed to pass the tract length), and in order to have a meaningful time interval for traffic flow simulation (Kotsialos and Papageorgiou, 2004). That is,

$T = 15$

$\Delta t = 10 \quad [s]$

$L_1 = 800 \quad [m]$

$L_2 = 800 \quad [m]$

$vel_1^t = 16.6 \quad [m/s]$

$vel_2^t = 16.6 \quad [m/s]$

Firstly, the optimization problem (1)-(7) has been solved, with the following inputs: $\underline{V} = [10,3,2,0,0,0,0,0,0,2,3,0,0,0,0]$, and the following weights in the objective function: $\alpha = 2 \cdot 10^4, \beta = 2 \cdot 10^4, \gamma = 2 \cdot 10^4, \delta = 2 \cdot 10^5$.

A receding-horizon control scheme has been applied and, in Table 1 and Table 2, the optimization results are reported.

Table 1: Results of the optimization problem: X^t, Z^t, I^t.

Time	X^t	Z^t	I^t
0	8.38	0	0
1	0.56	0	1.62
2	0.6	$0.26 \cdot 10^{-4}$	4.06
3	0.66	$0.26 \cdot 10^{-4}$	5.46
4	0.74	$0.25 \cdot 10^{-4}$	4.8
5	0.86	$0.24 \cdot 10^{-4}$	4.06
6	1	$0.23 \cdot 10^{-4}$	3.2
7	1.19	$0.21 \cdot 10^{-4}$	2.2
8	1	$0.19 \cdot 10^{-4}$	1
9	2	$0.17 \cdot 10^{-4}$	0
10	2.23	$0.14 \cdot 10^{-4}$	0
11	0.77	$0.12 \cdot 10^{-4}$	0.76
12	0	$0.83 \cdot 10^{-5}$	0
13	0	$0.44 \cdot 10^{-5}$	0
14	0	0	0

Table 2: Results of the optimization problem: N_1^t, N_2^t, Y^t.

Time	N_1^t	N_2^t	Y^t
0	0	0	0
1	$0.1 \cdot 10^{-1}$	0	1.74
2	$0.9 \cdot 10^{-2}$	$0.22 \cdot 10^{-2}$	1.49
3	$0.79 \cdot 10^{-2}$	$0.4 \cdot 10^{-2}$	1.3
4	$0.71 \cdot 10^{-2}$	$0.57 \cdot 10^{-2}$	1.17
5	$0.65 \cdot 10^{-2}$	$0.71 \cdot 10^{-2}$	1.08
6	$0.63 \cdot 10^{-2}$	$0.85 \cdot 10^{-2}$	0.04
7	$0.62 \cdot 10^{-2}$	$0.98 \cdot 10^{-2}$	1.03
8	$0.64 \cdot 10^{-2}$	$0.11 \cdot 10^{-1}$	1.06
9	$0.63 \cdot 10^{-2}$	$0.12 \cdot 10^{-1}$	1.05
10	$0.75 \cdot 10^{-2}$	$0.14 \cdot 10^{-1}$	1.25
11	$0.87 \cdot 10^{-2}$	$0.15 \cdot 10^{-1}$	1.45
12	$0.79 \cdot 10^{-2}$	$0.17 \cdot 10^{-1}$	1.31
13	$0.63 \cdot 10^{-2}$	$0.19 \cdot 10^{-1}$	1.04
14	$0.49 \cdot 10^{-2}$	$0.2 \cdot 10^{-1}$	0.82

The overall hazard is (summation over time of equation (6)) equal to 1978, with $\eta_{HAZ1}^t = \eta_{HAZ2}^t = \eta_{HAZ3}^t = 10$.
Then, the non-negativity constraints have been removed. The optimal values are the same like in the constrained case.

Similar results, in the unconstrained case, can be found through the use of the Riccati equation. Instead, for the constrained case an efficient method of solution has to be found. A possible approach can be the one reported in (Bertsimas and Brown, 2007). Otherwise, one can try to use dynamic programming and reduce the explosion of computation that arises.

6 CONCLUSIONS

A preliminary approach for the optimal control of hazardous materials traffic flow has been presented. The novelties of the presented approach in the literature of hazmat transportation have been highlighted, as well as the methodological approaches that might characterize the solution of the optimal control problem.

Future research related to the present work will regard the development of methods to derive the optimal control law to the considered problem in a closed form. After that, the decision problem could be extended to the optimal control of two fleets of hazardous material that have to flow through a tunnel in both competitive and collaborative cases. Moreover, a hierarchical control can be formalized in which a decision maker related to the tunnel has to decide the price to assign to the two fleets on the basis of the costs, the goods demand, and the risk to be minimized in the overall system, while the fleets aim at minimizing their own benefits and hazards.

REFERENCES

Bell, M., 2009. A multi-path Astar algorithm for risk averse vehicle navigation, *Transportation Research Part B: Methodological*, 43 (1), 97-107.

Bell, M., Cassir, C., 2002. Risk-averse user equilibrium traffic assignment: An application of game theory, *Transportation Research Part B: Methodological*, 36 (8), 671-681.

Berman, O., Verter, V., Kara, B.Y., 2007. Designing emergency response networks for hazardous materials transportation, *Computers and Operations Research*, 34(5), 1374-1388.

Beroggi, G., Wallace, W., 1994. Operational Risk Management: A New Paradigm for Decision Making, *IEEE Transactions on Systems, Man and Cybernetics*, 24 (10), 1450-1457

Eds Bersani, C., Boulmakoul, A., Garbolino, E., Sacile, R., 2008a*, Advanced Technologies and Methodologies for Risk Management in the Global Transport of Dangerous Goods,* NATO Science for Peace and Security Series - E: Human and Societal Dynamics (ISSN 1874-6276) Volume 45 ISBN 978-1-58603-899-1. Amsterdam: IOS Press.

Bersani, C., Minciardi, R., Sacile, R., Tomasoni, A., Trasforini, E., 2008b. An Integrated System for the Hazardous Materials Transport in a Sub-Regional Scale Area, in *Advanced Technologies and Methodologies for Risk Management in the Global Transport of Dangerous Goods*, Eds C.Bersani, A. Boulmakoul, E. Garbolino, R. Sacile, NATO Science for Peace and Security Series - E: Human and Societal Dynamics (ISSN 1874-6276) Volume 45 ISBN 978-1-58603-899-1. Amsterdam: IOS Press.

Bertsimas, D., Brown, D., 2007. Constrained Stochastic LQC: A Tractable Approach, *IEEE Transactions on Automatic Control*, 52 (10), 1826-1841.

Bonvicini, S., Spadoni, G., 2008. A hazmat multi-commodity routing model satisfying risk criteria: A case study, *Journal of Loss Prevention in the Process Industries* 21, 345–358

Castelein, R., Johnson, A., 1989. Constrained Optimal Control, *IEEE Transactions on Automatic Control*, 34 (I), 122-126

Fabiano, B., Currò, F., Palazzi, E., Pastorino, R., 2002. A framework for risk assessment and decision-making

strategies in dangerous good transportation, *Journal of Hazardous Materials* 93, 1–15.

Fabiano, B., Currò, F., Reverberi, A.P., Pastorino R., 2005. Dangerous good transportation by road: from risk analysis to emergency planning, *Journal of Loss Prevention in the Process Industries* 18, 403–413.

Kara, B., Verter, V., 2008. A Path-Based Approach for Hazmat Transport Network Design, Management Science 54 (1), 29-40.

Kara, B.Y., Verter, V., 2004. Designing a road network for hazardous materials transportation, *Transportation Science*, 38 (2), 188-196.

Kotsialos, A., Papageorgiou, M., 2004. Nonlinear optimal control applied to coordinated ramp metering, IEEE Transactions on Control Systems Technology 12 (6), 920-933.

Minciardi, R., Robba, M., Sacile, R., 2008. Traffic optimization in hazardous materials transport on roads flowing towards one critical road infrastructure, in *Advanced Technologies and Methodologies for Risk Management in the Global Transport of Dangerous Goods*, Eds C.Bersani, A. Boulmakoul, E. Garbolino, R. Sacile, NATO Science for Peace and Security Series - E: Human and Societal Dynamics (ISSN 1874-6276) Volume 45 ISBN 978-1-58603-899-1. Amsterdam: IOS Press.

Sadjadi, S.J., 2007. An application of efficient frontier in transportation of hazardous materials, *Computers & Industrial Engineering* 53, 357–360.

Serafini, P., 2006. Dynamic programming and minimum risk paths, *European Journal of Operational Research* 175, 224–237.

POSTERS

AN APPLICATION OF THE SPECTRAL KURTOSIS TO MONITOR CONTAINER GANTRY CRANES' MACHINERY

Juan José González de la Rosa, J. M. Sierra, A. Illana, J. A. Carmona, L. M. Calvente
University of Cádiz, Electronics Area, Research Group PAIDI-TIC-168, EPSA
Av. Ramón Puyol S/N, E-11202, Algeciras, Cádiz, Spain
juanjose.delarosa@uca.es

Antonio Moreno Muñoz
University of Córdoba, Electronics Area, Research Group PAIDI-TIC-168
Campus Rabanales, A. Einstein C-2. E-14071, Córdoba, Spain
amoreno@uco.es

Keywords: Fault detection, Gantry crane, Higher-Order Statistics, Spectral kurtosis, Transient detection, Vibration monitoring.

Abstract: The Spectral Kurtosis (SK) enhances non-Gaussian behavior associated to deviations from the nominal operation of the cranes machinery. This fact easies fault detection, with the subsequent prevention of dramatic malfunction. In this paper the rotor of a container gantry crane is monitored to get the kurtosis of its normal operation. Then, two types of rolling bearings faults are modeled, according to the design of the rotors crane. These signals are added to the real normal operation recordings, and processed under an estimator of the SK. The experience allows the conformation of a higher-order statistical fault-pattern data base, without the need of stopping huge machinery, and with the subsequent saving, settling the basis of an automatic surveillance system.

1 INTRODUCTION

The study of the vibrations in a gantry crane used in a containers' terminal is an issue related to the security of the crane operators and to the durability of the design. Vibrations take place mostly in the operator's cabin and in the machinery hall; see the photo in Fig. 1 to get a first approximation of the machinery under test.

Numerous achievements have been made in the field of the control for overhead crane systems, which have proven to be an improvement in the position accuracy, safety and stabilization control. To cite: (Ju et al., 2006; Hua and Shine, 2007; Lee et al., 2007). Furthermore, in the work (De la Rosa et al., 2007,), the cabin system has been modeled with *Simulink* and the vibration modes have been separated using the the independent component analysis, settling the basis of signal analysis in containers' cranes systems.

In the field of fault diagnosis, numerous improvements have been made, cataloguing faults within big machinery. The vast majority of the advances are based in the traditional power spectral analysis, which

Figure 1: Container Gantry Cranes at Algeciras harbor.

is very sensible to noise and does not offer a complete statistical characterization; in this sense, it is very well known the potential usability of Higher-Order Statistics (HOS) (De la Rosa and Muñoz, 2008,). Among them, it is worthy remarkable the improvement described in (Antoni, 2006) and (Antoni, 2007),

where the Spectral Kurtosis (SK) is applied to vibratory surveillance and diagnostics of rotating machines; faults are modeled herein and characterized using fourth-order statistics.

In this paper, the application of HOS consists of characterizing the normal operation of a crane's rotor. Then, based in this nominal kurtosis, we add two types of modeled faults to the normal operating recordings. These faults are associated to different catalogued defects in the rolling bearings (outer and inner race defects), in order to obtain their characterization based in the SK. The rotor is located in the machine hall of a container crane; so the described fourth-order analysis of the pseudo-synthetic signals, enables characterization without stopping the crane. Faults are modeled according to the dimensions of the rolling bearings.

An intermediate result proves the increment of the non-Gaussian feature of the faults. From the global calculation (numerical) of the kurtosis, we prove that the nominal operation is slightly non-Gaussian, and the kurtosis' increment is associated to faulty bearings. The main results will be concluded looking at the frequency patterns of the faults recordings, and they show the inter-frequency distance associated to both fault types.

The paper is structured as follows. In Section 2 we make a brief summary on the definition of kurtosis; we use an unbiased estimator of the SK, successfully used in (De la Rosa and Muñoz, 2008,), where a higher measurement bandwidth was used. Fault modeling is described in Section 3. Results are presented in Section 4. Finally, conclusions are drawn in Section 5.

2 KURTOSIS AND SPECTRAL KURTOSIS

In statistics, kurtosis is a measure of the "peakedness" of the probability distribution of a random variable X. Higher kurtosis means more of the variance is due to infrequent extreme deviations, as opposed to frequent modestly-sized deviations.

Kurtosis is more commonly defined as the fourth central cumulant divided by the square of the variance of the probability distribution, which is the so-called excess kurtosis, according to Eq. (1):

$$\gamma_2 = \frac{\kappa_4}{\kappa_2^2} = \frac{\mu_4}{\sigma^4} - 3, \tag{1}$$

where $\mu_4 = \kappa_4 + 3\kappa_2^2$ is the *4*th-order central moment; and κ_4 is the *4*th-order central cumulant, i.d. the ideal value of $Cum_{4,x}(0,0,0)$. This definition of the *4*th-order cumulant for zero time-lags comes from a combinational relationship among the cumulants of stochastic signals and their moments, and is given by the *Leonov-Shiryaev* formula. A complete description for these statistics can be found to cite in (Nikias and Mendel, 1993; Mendel, 1991).

The "minus 3" at the end of this formula is a correction to make the kurtosis of the normal distribution equal to zero. Excess kurtosis can range from -2 to $+\infty$.

A high kurtosis distribution has a sharper "peak" and fatter "tails", while a low kurtosis distribution has a more rounded peak with wider "shoulders". Distributions with zero kurtosis are called mesokurtic (e.g the normal distribution). A distribution with positive kurtosis is called leptokurtic. A leptokurtic distribution has a more acute "peak" around the mean and "fat tails" (e.g. the *Laplace* distribution). A distribution with negative kurtosis is called platykurtic, which has a smaller "peak" around the mean and "thin tails" (e.g. the continuous or discrete uniform distributions, and the raised cosine distribution; the most platykurtic distribution of all is the *Bernoulli* distribution).

In Measurement Science, the sample kurtosis is calculated over a sample-register (an *N*-point data record), and noted by:

$$g_2 = \frac{m_4}{s^4} - 3 = \frac{m_4}{m_2^2} - 3 = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^4}{\frac{1}{N^2}\left[\sum_{i=1}^{N}(x_i - \bar{x})^2\right]^2} - 3, \tag{2}$$

where m_4 is the fourth sample moment about the mean, m_2 is the second sample moment about the mean (that is, the sample variance), and $\bar{x}$ is the sample mean. The sample kurtosis defined in Eq. (2) is a biased estimator of the population kurtosis, if we consider a sub-set of samples from the population (the observed data).

In the frequency domain, the ideal SK is a representation of the kurtosis of each frequency component of a process (or data from a measurement instrument x_i) (De la Rosa and Muñoz, 2008; Vrabie et al., 2003,). For estimation issues we will consider M realizations of the process; each realization containing N points; i.d. we consider M measurement sweeps, each sweep with N points. The time spacing between points is the sampling period, T_s, of the data acquisition unit.

A biased estimator for the spectral kurtosis and for a number M of *N*-point realizations at the frequency index m, is given by Eq. (3):

$$\hat{G}_{2,X}^{N,M}(m) = \\ = \frac{M}{M-1}\left[\frac{(M+1)\sum_{i=1}^{M}|X_N^i(m)|^4}{\left(\sum_{i=1}^{M}|X_N^i(m)|^2\right)^2} - 2\right]. \quad (3)$$

This estimator is the one we have implemented in the program code in order to perform the data computation and it was also used successfully in (De la Rosa and Muñoz, 2008,). The estimator converges in probability to the quantity being estimated (the true value, γ_2) when the number of realizations M and the sample size N tend to $+\infty$. So we say that the estimator is asymptotically *consistent* with respect to M and N.

To show the ideal performance of the estimator, which has been described in these lines, and also described in (De la Rosa and Muñoz, 2008,), we show an example based in synthetics. A mix of six different signals have been designed. Each mixture is the sum of a constant-amplitude sine of 2 kHz, a constant-amplitude sine at 9 kHz, a Gaussian-distributed-amplitude sine al 5 kHz, a Gaussian-distributed-amplitude sine at 18 kHz, a Gaussian white noise, and a colored Gaussian noise between 12 and 13 kHz. Each mixture (realization or sample register) contains 1324 points.

Negative kurtosis ("-1") is expected for constant-amplitude processes, positive kurtosis (not bounded peaks) should be associated to random-amplitudes and zero kurtosis will characterize both Gaussian-noise processes. This is proved in Fig. 2, which shows a good performance because enough registers have been averaged (M=500).

The SK is supposed to behave similarly with synthetics associated to bearing fault modeling. As we find constant amplitude impulses (associated to faults), an infinite succession of "-1" should appear in the SK. This is also taken to measure inter-frequency distances.

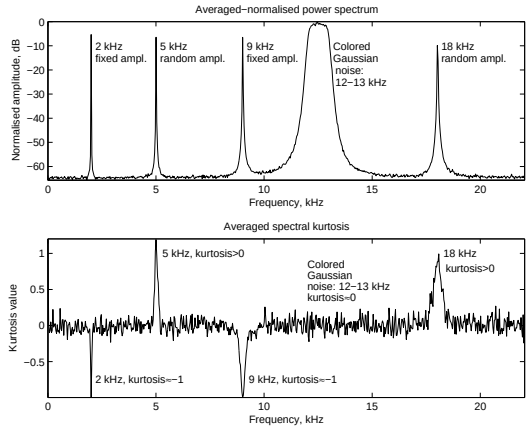


Figure 2: Performance over a set of synthetics, for M=500 realizations.

3 FAULT MODELING

Bearing components normally fail in the following order: race defects (the most common), ball or roller defects and cage defects (unless the bearing was defective when installed). Inner race defects and failures occur at much lower amplitudes than outer race defects.

BSF (Ball Spin Frequency) is usually generated when a ball or roller is defective. When multiple balls are defective, multiples of BSF appear, i.e., if BSF is at 800 RPM and four balls have defects, you should also see a peak at 3200 RPM or 4 $\times$ BSF. In all cases, a surface defect on an inner race, an outer race or on a roller (ball) generates shocks at the bearing characteristic frequencies.

In a frequency spectrum, defects correspond to pulse trains of frequencies extending from the 0-1000 Hz range in the domain of vibratory-acoustics. Such families of peaks merge with the peaks due to other causes. A real specialist must then deal with the bearing analysis to sort out other causes present in the frequency spectrum. In the vibration frequency range (typically 10-1000 Hz), the patterns of frequency spectra may indeed be complex, due to problems of rotor dynamics; pumps ventilators (blade passing frequencies, vanes, etc). This noise (usually Gaussian) can be rejected using HOS.

In the present work, we have modeled the outer race and the inner race faults, which are the most usual. Both faults are modeled with the same impulses' amplitudes. The following modeled magnitudes are exposed according to the jargon's nomenclature. The rotation speed (RPM/60), f = 1 Hz; BPFI (Bearing Inner Race Frequency) = 13 Hz; BPFO (Bearing Outer Race Frequency) = 7 Hz; BSF (Ball Spin Frequency) $\cong$ 1.15 Hz; FTF (Fundamental Train Frequency) $\cong$ 0.35 Hz.

The above magnitudes have been calculated considering the dimensions of the rolling bearing: contact angle (α) = zero; B_d (ball or roller diameter) = 65 mm; P_d (pitch diameter) = 215 mm; N_b (number of balls, or rollers) = 20. The following expressions expanded in Eq. (4), allow calculation for the present situation:

$$
\begin{aligned}
BPFO &= f \times \frac{N_b}{2} \times \left[1 - \frac{Bd}{Pd} \times cos(\alpha)\right] \\
BPFI &= f \times \frac{N_b}{2} \times \left[1 + \frac{Bd}{Pd} \times cos(\alpha)\right] \\
BSF &= (f/2) \times \frac{P_d}{B_d} \times \left[1 - \frac{Bd}{Pd} \times cos^2(\alpha)\right] \\
FTF &= (f/2) \times \frac{P_d}{B_d} \times \left[1 - \frac{Bd}{Pd} \times cos(\alpha)\right]
\end{aligned} \tag{4}
$$

According to Eq. (4), outer and inner faults have been modeled with the form of impulse trains whose pulse repetition rate obeys the calculation performed in Eq (4). Two sample registers are depicted in Fig. 3.

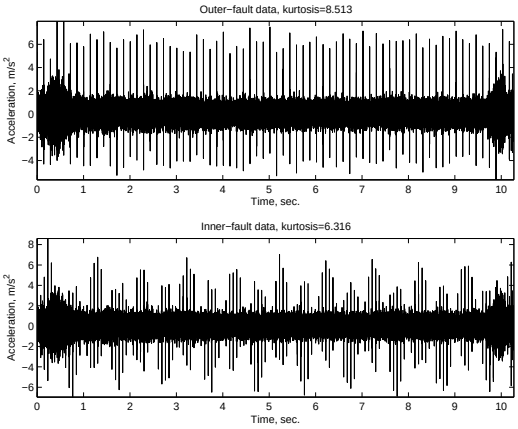


Figure 3: Two data registers which model both types of faults in the rotor's rolling bearings. Impulse repetition: BPFI (Bearing Inner Race Frequency) = 13 Hz; BPFO (Bearing Outer Race Frequency) = 7 Hz.

It is also seen in Fig. 3, the background raw data, which corresponds to normal operation. A SNR of 6 dB has been fixed (variance of the normal vibration, 4; variance of the pulse trains 8). According to this model, results for the SK analysis are presented hereinafter.

4 EXPERIMENTS AND RESULTS

We show the experimental location of the sensors in Fig. 4. Three sensors are primarily connected to the rotor carcase, aiming to confirm the similarity of the three signals, in order to reduce the three to one measurement point. The rotor's structure under test is located inside the machine room of the crane (see Fig. 1).

Industrial accelerometers (model KD42V) have been used with a sensitivity of 100 mV/g, which is the usual standard in noise and vibration control Engineering. The sensors' usable bandwidth is of 100 kHz. Despite the fact that low-frequency vibrations are involved, the sampling frequency was set to 10 kHz in order to capture high resolution recordings, aiming to buried them with modeled impulses' trains.

Figure 4: A photograph of the rotor and the location of the sensors.

The kurtosis as a global indicator, considered as the average of the kurtosis computed for each individual frequency component, is not a valid tool to extract features. This is due to the fact that no discrimination is made neither among the frequency bands nor the frequency pattern, from the global point of view.

In order to get a reliable characterization, each register (10 kHz sampled) contains numerous data (102,714; about 10 sec. sampling). In normal operation, the mean value of the excess kurtosis is 1.343, which is somewhat over the Gaussian limit (kurtosis = 0). For the outer fault case, the median of the kurtosis is 8.513. The inner fault is characterized by an average kurtosis of 6.316. This by the way is an indication of the type of fault. But the global indication is very susceptible to errors associated to transients, or other non-Gaussian noise, and does not provide information relative to the frequency bands.

So, the key of the SK detection strategy used in this work, lies in the potential enhancement of the non-Gaussian behavior of the vibrations. If this happens, i.e. if an increase of the non-Gaussian activity (increase in the kurtosis, peakedness of the probability distribution) is observed-measured in the SK graph, there may be deviation from normal functioning.

Fig. 5 shows the frequency analysis associated to one recording, which models an inner race defect in the rotor bearing. At a first glance, it is difficult to reach a frequency pattern, but a closer examination reveals the constant inter-frequency distance which

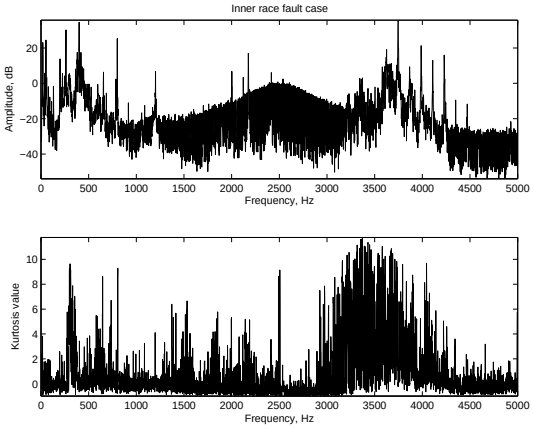


Figure 5: High resolution frequency analysis of an inner race fault.

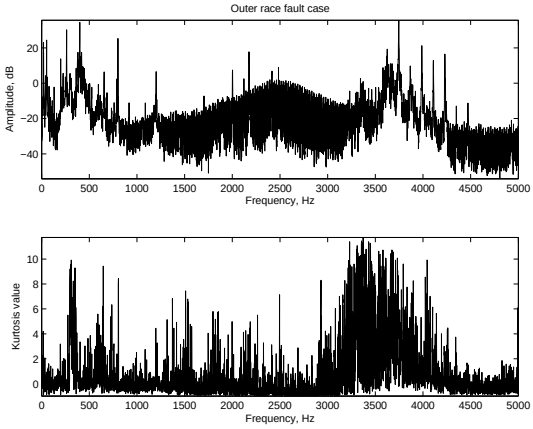


Figure 7: High resolution frequency analysis of a fault in the outer race.

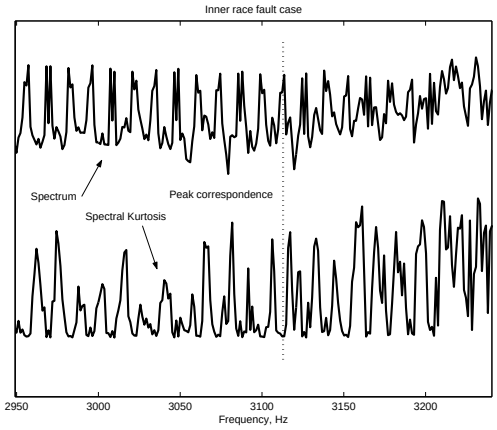


Figure 6: High resolution frequency analysis of an inner race fault. A zoom of Fig. 5. Graphs have been shifted for convenience. Inter-frequency = 13 Hz.

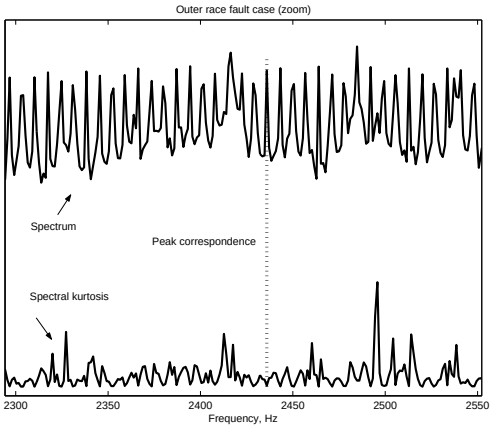


Figure 8: High resolution frequency analysis of an outer race fault. A zoom of Fig. 7. Graphs have been shifted to improve visualization. Inter-frequency = 7 Hz.

characterizes this type of fault.

Fig. 7 presents the spectral analysis of an outer race fault, which is very similar to the inner fault case, depicted in Fig. 6.

On the basis of the second and fourth-order spectra, we conclude the possibilities of the SK to distinguish between two common faults in rotor bearings.

5 CONCLUSIONS AND ACCOMPLISHMENTS

Results show the potential use of the SK to target faults in mechanical systems. Concretely, the estimator of the SK is able to discriminate between two different faults, commonly encountered in rolling bearings, and targeted here via the inter-frequency distance.

The improved performance of the SK over the global excess kurtosis resides in the possibility of analyzing separated frequency bands, or inter-frequency distances, which are more indicative features of faults than a mere numeric statistical calculation. The kurtosis as a global indicator is considered only a prior indication of the fault.

The interest of the experiment resides in the possibility of incorporate this signal processing algorithm to the engine of an expert system in order to monitor on-site performance of machinery, and get a predictive surveillance. This would be done without stopping production of such big machinery.

REFERENCES

Antoni, J. (2006). The spectral kurtosis: application to the vibratory surveillance and diagnostics of rotating machines. *Mechanical Systems and Signal Processing (Ed. Elsevier)*, 20(2):308–331.

Antoni, J. (2007). Cyclic spectral analysis in practice. *Mechanical Systems and Signal Processing (Ed. Elsevier)*, 21(2):597–630.

Hua, Y. J. and Shine, Y. K. (2007). Adaptive coupling control for overhead crane systems. *Mechatronics*, (-):in Press.

Ju, F., Choo, Y., , and Cui, F. (2006). Dynamic response of tower crane induced by the pendulum motion of the payload. *International Journal of Solids and Structures*, (43):376–389.

Lee, D.-H., Cao, Z., and Meng, Q. (2007). Scheduling of two-transtainer systems for loading outbound containers in port container terminals with simulated annealing algorithm. *Int. J. Production Economics*, (-):in Press.

Mendel, J. M. (1991). Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications. *Proceedings of the IEEE*, 79(3):278–305.

Nikias, C. L. and Mendel, J. M. (1993). Signal processing with higher-order spectra. *IEEE Signal Processing Magazine*, pages 10–37.

De la Rosa, J. J. G. and Muñoz, A. M. (2008). Higher-order cumulants and spectral kurtosis for early detection of subterranean termites. *Mechanical Systems and Signal Processing (Ed. Elsevier)*, 22(Issue 1):279–294. Available online 1 September 2007.

De la Rosa, J. J. G., Puntonet, C. G., Moreno, A., Illana, A., and Carmona, J. A. (2007). An application of ICA to BSS in a container gantry crane cabin's model. *Lecture Notes in Computer Science (LNCS)*, 4666:714–721. 7th International Conference on Independent Component Analysis and Signal Separation (ICA 2007) London, UK, September 9-12, 2007, Proceedings.

Vrabie, V., Granjon, P., and Serviere, C. (2003). Spectral kurtosis: from definition to application. In IEEE, editor, *IEEE-EURASIP International Workshop on Nonlinear Signal and Image Processing (NSIP'2003)*, volume 1, pages 1–5.

A PARAMETERIZABLE HANDEL-C NEURAL NETWORK IMPLEMENTATION FOR FPGA

Cherrad Benbouchama[1], Mohamed Tadjine[2]
[1] *E.M.P, Bordj El Bahri, 16111 Alger, Algeria*
[2]*Ecole Nationale Polytechnique, El Harrach Alger, Algeria*
ben_cherrad@yahoo.fr

Ahmed Bouridane[3]
[3]*School of Electronics, Electrical Engineering and Computer Science, Queen's University*
BT7 1NN, Belfast, U.K.

Keywords: Neural networks, FPGA, Parameterizable implementation, Handel-C.

Abstract: This paper shows the design possibility of a parameterizable implementation of neural multi-layer network on FPGA circuits (Field Programmable Gate Array) through the use of Handel-C language. The algorithm used for the training is the back-propagation. The tools of implementation and synthesis are the DK 4 of Celoxica and the ISE 6.3 of Xilinx. The targeted components are XCV2000 on Celoxica RC1000 board and XC2V1000 on RC200. The representation of the real numbers in fixed point was used for the data processing. The realization of the activation function is made with the approximate polynomial. A high level environment was designed in order to specify and introduce architecture parameters.

1 INTRODUCTION

The first ANNs implementation on FPGA was carried out by Cox and al within the framework of the GANGLION project (Cox and Blanz, 1992). It was applied for real time images segmentation. Other work followed since, for the implementation of various types' architectures of ANNs and various real data representations. The algorithm most used for multi-layer ANNs training is the back-propagation one. Eldredge had made a success, in 1994, of the first implementation of this algorithm on the RRANN platform (*Runtime Reconfiguration Artificial Neural Network*) built around the XC3090 FPGA (Eldredge and Hutchings, 1994). Ferrucci and Martin have designed the ACME multi-FPGA platform *(Adaptive Connectionist Model Emulator*) which is composed of fourteen (14) *Xilinx* XC4010 FPGAs (Ferrucci 1994) (Martin, 1994). Ossoing, in 1996, had also implemented on FPGA the back-propagation algorithm. It had implemented the architecture [3,3,1] on four (4) Xilinix XC4013 FPGAs and one Xilinix XC4005 (Ossoinig and al, 1996). Beuchat & al., in 1998, were heavily influenced by Eldredge work. They developed the RENCO platform containing four (4) FPGAs controlled by microprocessor. It was successfully applied to hand-written characters recognition (Beuchat and al, 1998). Pandya, in 2005, had implemented on FPGA the back-propagation algorithm with Handel-C language and had worked out a partially parallel architecture and another completely parallel, which proved, respectively, twice and four times faster than a sequential architecture (Pandya, 2005). However, the necessary time for the ANNs implementation on FPGAs is of about months (Benbouchama and al, 2007). Moreover, the programming of these circuits requires the mastery of specific languages, which reduces considerably their use on a large scale. It would be then interesting to carry out a graphical environment which would be dedicated for users not necessarily initiated to FPGAs programming.

In this paper, we are interested in the design of a parameterizable tool that generates a neural multi-layer network implementation on FPGAs through the use of Handel-C language. This work aims at the realization of a high-level environment making it possible, on the one hand, to ensure the interfacing between the user and an FPGA board, and on the other hand, to generate automatically configurations dedicated to the ANNs. The algorithm used for the

training is the back-propagation, and the tools of implementation and synthesis are the DK 4 of Celoxica and the ISE 6.3 of Xilinx.

2 TECHNICAL PRINCIPLE

2.1 Handel-C Language

Handel-C is perhaps the most popular high-level language currently available for hardware specification (Stöcklein and Bhig, 2002). Its syntax is based on the C language making it easily adopted by traditional software engineers. The benefits of rapid hardware development and simplicity of specification often come at a price. In comparison with traditional hardware description languages such as VHDL (Ashenden, 2002), Handel-C and its compiler produce hardware that consumes more FPGA area (Hopf, 2003) and is often slower in performance.

2.2 A Parameterizable Implementation

A parameterizable implementation is that in which a user would have the possibility of changing the application parameters without having to modify the hardware configuration of the FPGA. The implementation will be flexible.

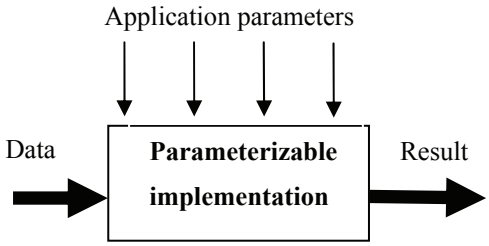


Figure 1: A parameterizable implementation.

In a parameterizable ANNs implementation (figure 2), the inputs are, in addition to the maximum of epochs and the desired error, different data concerning the architecture and the training parameters.

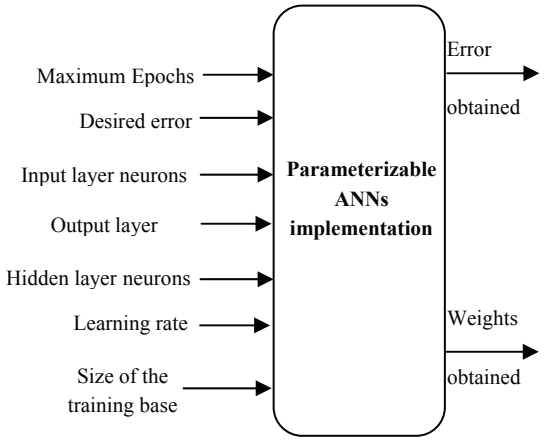


Figure 2: Parameterizable ANNs implementation.

The development of this parameterizable ANNs implementation is made with handel-C language which, contrary to VHDL, makes it possible to design completely parameterizable applications.

2.3 Automatic Generation of Configurations

It consists in developing an automatic generator of configurations starting from the host computer towards the FPGA board (figure 3).

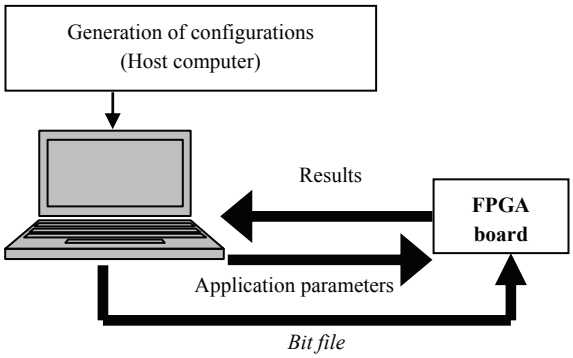


Figure 3: Principle of the software platform.

The generator of configurations ensures the following tasks:

- The synchronization between the host computer and the FPGA board.
- The configuration of the FPGA circuit by the bit file.
- The data transfer between the FPGA board and the host computer.

2.4 Graphical Interface

From these graphical menus, the user is invited to design his application. Thus, he can specify the different parameters of the application. This system can bring many tools for the implementation design and control. The advantage resides in the facility of the approach: there is no language, it requires only the use of graphical menus (control buttons, edition zone…). The graphical environment carried out with *Microsoft visual C++* can be composed of one or several windows.

3 IMPLEMENTATION RESULTS

To demonstrate the findings, the parameterizable implementation was realized using the XCV2000 of the FPGA family VirtexII.

3.1 First Case

Input layer neurons number ≤ 2
Output layer neurons number ≤ 2
Hidden layer neurons number ≤ 15
Hidden layer number ≤ 3

Table 1: First case.

Resources	Used	Available	Utilization
Slices	17019	19200	88 %
LUTs	31320	38400	81 %
IOBs	88	404	21 %
Block RAMs	320	160	**200 %**

3.2 Second Case

Input layer neurons number ≤ 2
Output layer neurons number ≤ 2
Hidden layer neurons number ≤ 10
Hidden layer number ≤ 1

Table 2: Second case.

Resources	Used	Available	Utilization
Slices	13101	19200	68 %
LUTs	24978	38400	65 %
IOBs	88	404	21 %
Block RAMs	180	160	**112 %**

3.3 Third Case

Input layer neurons number = 1
Output layer neurons number = 1
Hidden layer neurons number ≤ 10
Hidden layer number ≤ 1

Table 3: Third case.

Resources	Used	Available	Utilization
Slices	12144	19200	63 %
LUTs	23223	38400	60 %
IOBs	88	404	21 %
Block RAMs	108	160	67 %

From these results, it can be stated that the size of the parameterizable implementation to be realized depends on the targeted FPGA.

4 EXPERIMENTAL EVALUATION: THE POSITION CONTROL OF A DC MOTOR

We test the high level-environment on the implementation of a neural controller which improves an existing linear controller in order to position control a DC motor (Fig. 4). This approach is called "feed-back error learning" and is based on the use of an existing regulator to approximate as an unknown function (Benbouchama and al, 2007).

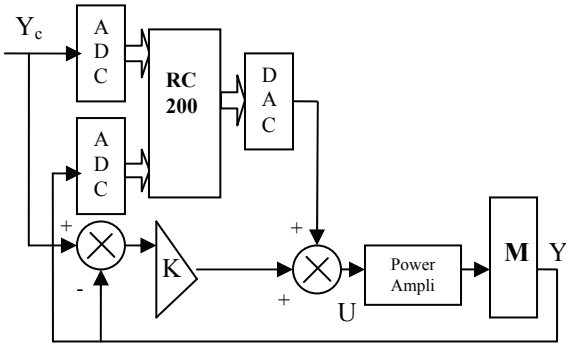


Figure 4: The position control of a DC motor.

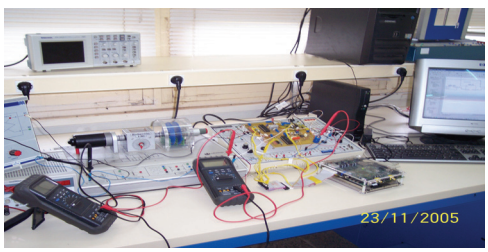


Figure 5: Hardware view.

In this experiment an on-chip learning type of the neural controller was used.
The graphical interface used to specify the different parameters of the neural controller is as follows:

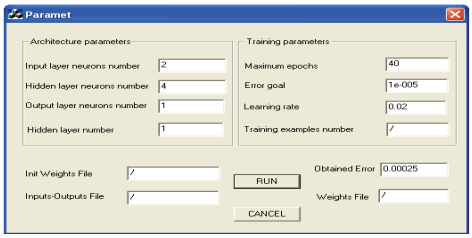


Figure 6: The graphical interface.

The implementation results of the parameterizable ANNs implementation for this experimental evaluation are as follows:

Table 4: Implementation results.

Resources	Used	Available	Utilization
Slices	2748	5120	53 %
LUTs	4926	10240	48 %
IOBs	20	324	6 %

Once the learning phase was completed we have tested the behaviour of the neural controller with a square input signal: 0° - 45°.

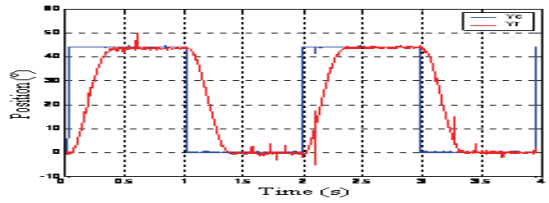


Figure 7: The behaviour of the neural controller.

The results indicate the validity of the high-level environment used for the ANNs implementation on FPGAs.

5 CONCLUSIONS

This paper outlines a means that was created to facilitate and accelerate the ANNs implementation on FPGAs. A parameterizable tool was designed to generate a neural multi-layer network implementation through the use of Handel-C language. This tool was destined to be used by the high level environment, which is presented, at the user, as a graphical interface with menus. The advantage which it offers resides in the facility of the approach: there is no language and it requires only the use of the graphical menus.
To be able to implement significant neural networks architectures with this high level environment, we must use a board with an FPGA circuit which is not limited in resources.
Finally, experimental evaluation setup has been developed to demonstrate the validity of the high-level environment for the ANNs implementation on FPGAs.

REFERENCES

Cox, C.E. and E. Blanz, GangLion, " a fast field - programmable gate array implementation of a connectionist classifier ", *IEEE Journal of Solid-State Circuits, 1992. 28(3): p. 288-299.*

J. G. Eldredge and B. L. Hutchings, " RRANN: A Hardware Implementation of the Backpropagation Algorithm Using Reconfigurable FPGAs ", *In IEEE World Conference on Computational Intelligence, Orlando, FL, 1994.*

A. T. Ferrucci, " A Field-Programmable Gate Array Implementation of Self-Adapting and Scalable Connectionist Network ", *Mars 1994.*

M. Martin, *"A reconfigurable hardware accelerator for back-propagation connectionist classifiers"*, Masters thesis, University of California, Santa Cruz, 1994.

H. Ossoinig, E. Reisinger, C. Steger, and R. Weiss, "Design and FPGA implementation of a neural network ", *In Proc. 7th Int. Conf. on Signal Processing Applications and Technology, pp. 939943, Orlando, Florida, 1996.*

J. Beuchat, J. Haenni, and E.Sanchez, *" Hardware Reconfigurable Neural Networks, In Parallel and Distributed Processing "*, IPPS/SPDP, pp. 91 98, Springer-Verlag, 1998.

V. Pandya, *" A Handel-C implementation of thebackpropagation algorithm on field programmable gate arrays "*, Master thesis. Faculty of Graduate Studies of the University of Guelph, Canada, December 2005.

C. Benbouchama and al, *" The FPGA Neural Networks Implementation for a Real Time Control "*, Archives of Control Sciences (A.C.S.), Vol. 17(LIII), n. 1, pp. 527, 2007.

T. Stöcklein and J. Bhig, *" Handel-C an effective method for designing FPGAs (and ASICs) ",* Georg Simon-Ohm Fachhochschule, NÜRNBERG 2002.

P. J. Ashenden, *" The Designer's Guide To VHDL "*, second edition, San Diego: Morgan Kanfmann, 2002.

J. Hopf, " Comparing the Bitstreams of Applications Specified in Hardware Join Java and HandelC ", *2nd IEEE International Conference on Field Programmable Technology, Tokyo, Japan, 2003.* Celoxica: www.Celoxica.com

HIERARCHICAL PERFORMANCE-ORIENTED CONTROL OF FLEXIBLE MANUFACTURING CELLS

Sherif Fahmy, Subramaniam Balakrishnan and Tarek ElMekkawy
Department of Mechanical and Manufacturing Engineering, University of Manitoba
75A Chancellors Circle, Winnipeg, Manitoba, R3T5V6, Canada
{umfahmy, balakri, tmekkawy}@cc.umanitoba.ca

Keywords: Deadlock-free scheduling, Flexible manufacturing cells, Hierarchical control, Job shops, Marked graphs, Petri nets.

Abstract: In a job shop, each product may have a different processing route through the system. Automated Flexible Manufacturing Cells (FMC) that adopt this flow pattern are highly prone to deadlocks. A supervisor is a controller that uses available data via feedback loops to characterize the current behavior of the cell, and modify the equipment controllers to achieve the desired operational specifications in a deadlock-free manner. This paper proposes a hierarchical control system divided into an upper level scheduler and a lower level supervisor to control FMCs. The scheduler is responsible for determining a deadlock-free allocation of the resources that optimizes some performance measure, based on the current production requirements, and the supervisor guarantees that the flow plan (behavior) determined by the scheduler is realized on the shop floor. For that purpose, a formal method that can transform a production schedule into a supervisor, in real time, is also proposed. The supervisor is an augmented Marked Graph (MG) that captures all the events that can take place in the cell. The proposed approach is validated by generating and simulating the supervisors for two benchmark problems.

1 INTRODUCTION

Most automated manufacturing systems (AMSs) feature three inherent operational properties; mutual exclusion, no pre-emption, and the hold-while-wait property. Because of these conditions and the inherent flow complexities in job shop systems, when they are automated they become highly prone to deadlocks. A *deadlock* occurs in an automated manufacturing system when a set of jobs enter a *circular wait*, where each job continues holding (blocking) a system resource indefinitely while waiting for another resource to become available, which is in turn held by another job in this same set.

Scheduling and control of manufacturing systems have been widely researched and reported in literature in the past decades. However, a wide gap exists between the contributions found in the scheduling literature and those pertaining to actual implementation (supervision) on the shop floor (Sun et al., 2006). A few attempts, however, have been made to integrate deadlock-free scheduling and supervision, but these either lacked a global view of the system (Li & Jiang, 2006), or realized a poorer performance when compared to pure deadlock-free scheduling approaches.

In the previous literature, the Supervisory Control Theory (SCT) (Ramadge & Wonham, 1987) and Petri nets (PNs) have been the two most frequently used and commonly accepted methods by researchers for modeling and supervising AMSs. Limitations of the SCT approaches have been attributed to the large state space required to represent even small systems, and the complexity of analysis of the formal languages. On the other hand, PN literature on Supervisory Control (SC) can be classified into approaches that analyzed the Reachability graph of the net (Viswanadham et al., 1990, Hsieh & Chang, 1994) and approaches that characterized the deadlock states using siphons analysis (Ezpeleta et al., 1995, Chu & Xie, 1997). While the former approaches suffered either from the state explosion problem or the restrictiveness of the PN model (Fanti & Zhou, 2004), the latter ones suffered from the exponential complexity of determining the siphons of the net.

Automata and PN SC approaches have usually been combined with conventional scheduling

approaches to solve the deadlock-free scheduling problem (Liljenvall, 1999, Golamakni et al., 2006, Lee & DiCesare, 1994, Ben Abdallah et al., 2002). However, these approaches have suffered from the same complexities which their SC counterparts have suffered from. To cope with these complexities, some heuristic approaches have been introduced to the literature to solve the problem. These included the work proposed in Huan & Wu (2004), Mati et al. (2001), and Fahmy et al. (2008). Others proposed mathematical formulations that can be solved to attain the optimal solutions for the problem (Ramaswamy & Joshi, 1996).

Nevertheless, the literature still lacks a formal approach that can transform a deadlock-free schedule of a job shop system into an implementable supervisor. The existence of such an approach would guarantee the correct and performance-optimized behavior of the system.

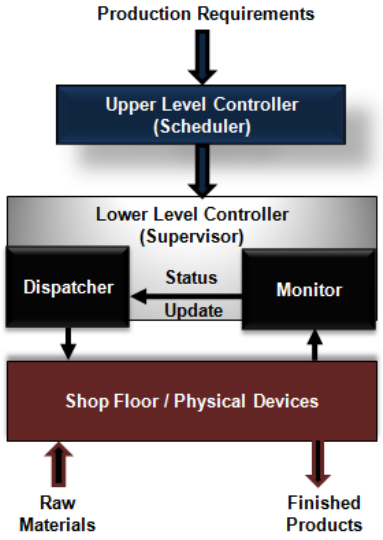


Figure 1: Proposed hierarchical control system.

2 HIERARCHICAL CONTROL

The type of systems considered is Flexible Manufacturing Cells (FMCs) that feature a job shop flow pattern. Such cells usually comprise a number of CNC machines that are served by a dedicated material handler (like a robot manipulator). In addition, they usually include some buffer capacity that can be used to temporarily store a job to preserve the continuity of flow or to resolve a deadlock.

Ideally, the functions of a production control system can be classified into three distinct functional modules; a scheduler, a monitor, and a dispatcher. Accordingly, a hierarchical control system divided into an upper level scheduler, and a lower level supervisor that monitors and dispatches commands to the shop floor (Figure 1) is proposed. According to the current product mix, the scheduler allocates processing slots for the jobs on the available machines while optimizing an objective criterion. The schedule further ensures that the resulting job flow cannot cause any deadlock situations. The assigned processing slots, and hence the underlying flow plan of the schedule is *transformed* into a supervisor. The supervisor will guarantee that the flow plan (behavior) determined by the scheduler is realized on the shop floor. The supervisor then interacts with shop floor devices by receiving feedback signals and accordingly issuing action commands directly from/to the shop floor.

3 SUPERVISOR REALIZATION

PN supervisors embedding a Marked Graph (MG) structure can be easily verified for liveness and reversibility. A MG is an ordinary PN in which each place has exactly one input transition and one output transition. A MG is live (deadlock-free) if the net structure obtained by deleting all the places marked by the initial marking contains no circuits, and a live MG is also reversible (Campos et al., 1992). Accordingly, the proposed approach initially transforms a given deadlock-free schedule into a live and reversible MG.

Consider the schedule of three jobs on three machines shown in Figure 2.

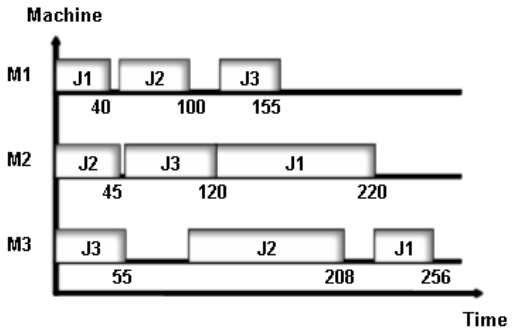


Figure 2: Schedule of illustrative example.

The first step to transform such schedule into a MG is to represent the processing route of each job by a production Petri net (PPN) (Banaszak & Krogh, 1990). This PPN provides the sequence of places

and transitions that describe the flow of the job; places represent the processing operations, and transitions model the release and/or acquisition of the corresponding machine(s). A token in these places (*flow places)* indicates that a job is currently holding the corresponding machine, either while begin processed or while waiting for the next machine in its route. To represent the sequence of jobs visiting each machine as indicated by the schedule, each transition representing the release of a machine is connected to an additional place (*scheduling place*) by an input arc. This place is then connected by an output arc to the transition representing the acquisition of the same machine by the next visiting job. Accordingly, this machine will not be assigned to the next job until it is released by the current job (hold-while-wait condition). In order to ensure the initiation and repetition of the schedule, a token-occupied place is added between the transition that releases the machine from the last job in the visiting sequence, and the transition that acquires the machine for the first job in the sequence (Figure 3). Note that the resulting net is still a MG and will henceforth be referred to as a *scheduling marked graph* (SMG).

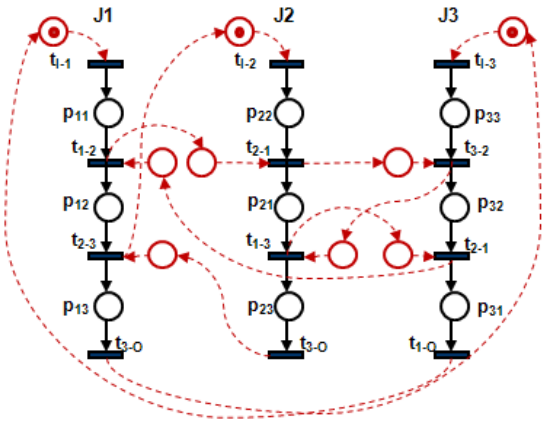


Figure 3: SMG of illustrative example.

The schedule shown in Figure 2 features a circular wait that would eventually result in a deadlock (Figure 3 contains three empty circuits). This circular wait can be resolved by placing J_1 in the buffer after completion on machine M_1, and hence expanding $t_{1\text{-}2}$ into a flow place p_{1B} in-between two new transitions, $t_{1\text{-}B}$ and $t_{B\text{-}2}$. A token in p_{1B} represents J_1 while residing in the buffer. Firing $t_{1\text{-}B}$ releases M_1 and places J_1 in the buffer, while firing $t_{B\text{-}2}$ acquires M_2 and moves J_1 from the buffer.

Using a hybrid approach earlier proposed in literature (DiCesare et al., 1993), through a series of top-down and bottom-up steps, the obtained SMG can then be augmented to represent the material handling (robot) operations while preserving the liveness and reversibility of the original SMG. Top-down decomposition first divides each flow place into two places with a transition in-between. The first place models the robot while handling the job and the second preserves the function of the original flow place. In order to ensure that the robot is not acquired simultaneously by more than one job, the bottom-up aggregation step adds a robot place p_R with one token to the SMG. This place is connected with output arcs to transitions that model the acquisition of the robot, and input arcs from transitions that model its release. After applying both steps, the augmented SMG (ASMG) can be obtained as shown in Figure 4. In this figure, arcs that connect p_R to its associated transitions are partially omitted and scheduling places along with their corresponding arcs are represented by bold arcs for the sake of clarity.

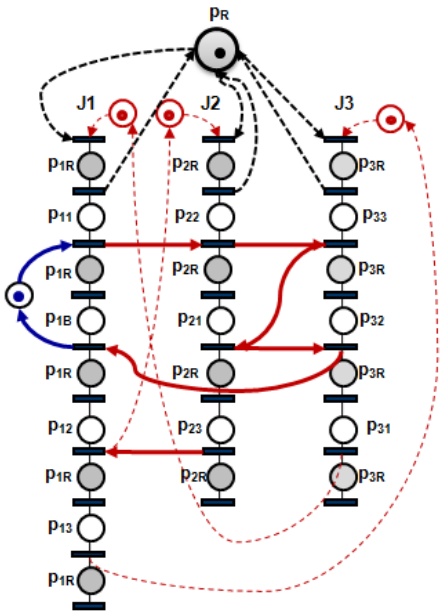


Figure 4: ASMG of illustrative example.

4 APPROACH VALIDATION

In order to validate the proposed approach, the supervisors for two benchmark problems are generated and simulated. Simulation entails executing the corresponding ASMGs of the problems to simulate the production process. The selected problems are the '4 jobs x 3 machines' problem introduced in Ramaswamy & Joshi (1996),

and a '6 jobs x 6 machines' problem that can be found in the OR library under the name *ft06*. The instance selected for problem '4J x 3M' features a unit buffer capacity, and for problem ft06, no buffer space is available in the system. The times required to obtain the deadlock-free schedule using the heuristic proposed in Fahmy et al. (2008) and generate the corresponding ASMG for the 4Jx3M and *ft06* problems were 0.19 and 0.8 seconds, respectively. In order to test the reversibility of the supervisors, they were run for lot sizes of five parts for each job type. The two ASMGs were executed, and all the parts for all the job types for the two problems were completed successfully. The two ASMGs can now be implemented through a computer, which can be connected to cell devices to complete the required product mixes.

5 CONCLUSIONS

This paper has proposed an efficient hierarchical scheduling and control architecture for FMCs. The inputs to the proposed architecture are simply the available resources in the system and the production routes of the jobs to be produced. The output is a readily implementable supervisor, capable of driving the system to autonomously produce the required products in a deadlock-free manner, according to the best production schedule. The supervisor can further be updated in real time to accommodate any changes in the product mix, while preserving the optimized performance of the system. The output of this work can to some extent narrow the gap that exists between scheduling and control literature of AMSs.

REFERENCES

Banaszak, Z. A., Krogh, B. H., 1990. Deadlock avoidance in flexible manufacturing systems with concurrently competing process flows. *IEEE Transactions on Robotics and Automation,* vol. 6, no. 6, pp. 724-734.

Ben Abdallah, I., ElMaraghy, H. A., ElMekkawy, T., 2002. Deadlock-free scheduling in flexible manufacturing systems using Petri nets. *International Journal of Production Research*, vol. 40, no. 12, pp. 2733-2756.

Campos, J., Chiola, G., Colom, J. M., Silva, M., 1992. Properties and performance bounds of timed marked graphs. *IEEE Transactions on Circuits and Systems*, vol. 39, no.5, pp. 386-401.

Chu, F., Xie, X.-L., 1997. Deadlock analysis of Petri nets using siphons and mathematical programming. *IEEE Transactions on Robotics and Automation,* vol. 13, no. 6, pp. 793-804.

DiCesare, F., Harhalakis, G., Proth, G. M., Silva, M., Vernadat, F. B., 1993. Practice of Petri nets in manufacturing. *Chapman & Hall*, London.

Ezpeleta, J., Colom, J. M., Martinez, J., 1995. A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE Transactions on Robotics and Automation,* vol. 11, no. 2, pp. 173-184.

Fahmy, S. A., ElMekkawy, T. Y., Balakrishnan, 2008. Deadlock-free scheduling of flexible job shops with limited capacity buffers. *European Journal of Industrial Engineering*, vol. 2, no. 3, pp. 231 – 252.

Fanti, M. P., Zhou, M., 2004. Deadlock control methods in automated manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 34, no.1, pp. 5-32.

Golmakani, H. R., Mills, J. K., Benhabib, B., 2006. Deadlock-free scheduling and control of flexible manufacturing cells using automata theory. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 36, no.2, pp. 327-337.

Hsieh, F.-S., Chang, S.-C., 1994. Dispatching-driven deadlock avoidance controller synthesis for flexible manufacturing systems. *IEEE Transactions on Robotics and Automation,* vol.10, no. 2, pp. 196-209.

Lee, D. L., DiCesare, F., 1994. Scheduling flexible manufacturing systems using Petri nets and heuristic search. *IEEE Transactions on Robotics and Automation,* vol.10, no. 2, pp. 123-132.

Li, L., Jiang, Z., 2006. Formal design and analysis of a hybrid supervisory control structure for virtual production systems. *International Journal of production Research*, vol. 44, no.13, pp. 2479-2497.

Liljenvall, T., 1999. Scheduling for production systems with limited buffers. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 6, pp. 469-474.

Mati, Y., Rezg, N., Xie, X., 2001. Geometric approach and Taboo search for scheduling flexible manufacturing systems. *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 805-818.

Ramadge, P. J., Wonham, W. M., 1987. Supervisory control of a class of discrete event processes. *Siam J. Control and Optimization*, vol. 25, no. 1, pp. 206-230.

Ramaswamy, S. E., Joshi, S. B., 1996. Deadlock-free schedules for automated manufacturing workstations. *IEEE Transactions on Robotics and Automation,* vol. 12, no.3, pp. 391-399.

Sun, R.-L., Li, H.-X., Xiong, Y., 2006. Performance-oriented integrated control of production scheduling. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 36, no. 4, pp.554-562.

Viswanadham, N., Narahari, Y., Johnson, T. L., 1990. Deadlock prevention and deadlock avoidance in flexible manufacturing systems using Petri net models. *IEEE Transactions on Robotics and Automation,* vol.6, no. 6, pp. 713-723.

INTELLIGENT HIERARCHICAL CONTROL SYSTEM FOR COMPLEX PROCESSES

Three Levels Control System

Yuri V. Mitrishkin
Bauman Moscow State Technical University, Second Baumanskaya St.,5, Moscow, Russia
y_mitrishkin@hotmail.com

Rodolfo Haber Guerra
Instituto de Automática Industrial, CSIC, Madrid, Spain
rhaber@iai.csic.es

Keywords: Complex dynamic systems, Feedback, Hierarchical, Robust, Adaptive, Self-organizing, Decision making, Intelligent control, Plasma in tokamaks.

Abstract: The paper presents a concept of intelligent hierarchical control for complex dynamical processes and suggests architecture of control system consisting of robust, adaptive, and self-organizing levels. Intelligent features of the proposed system are mostly concentrated at self-organizing level incorporated into self-learning, self-configuring, self-optimizing, and decision making algorithms. State-of-the-art at each level is described. Case studies have been chosen from the area of plasma control in tokamak-reactors.

1 INTRODUCTION

Recent advances in control strategies, communications, hard and soft-computing technologies have favoured an increasing trend towards the new generation of networked control systems for complex processes. The proposal described herein will address the development of scalable control methods and systems in accordance with the Information and Communication Technologies (ICT) Work Programme, ICT-2009 3.5a: Foundations of complex systems engineering: To achieve robust, predictable and self-adaptive behavior for large-scale networked systems characterized by complex dynamic behavior through the development of novel abstractions and scalable methods for sensing, control and decision-making. The scope covers foundational multi-disciplinary research and proof of concept addressing the whole chain from modeling, sensing, monitoring and actuation, to adaptive and cooperative control and decision making (European Commission, 2008).

To meet the goal stated by the EC a three level intelligent hierarchical control system was suggested by Bauman Moscow State Technical University to be applied to solve control problems of complex dynamic processes in science, engineering, and industry.

The project is focused on design and development of scalar (Single-Input/Single-Output: SISO) and multivariable (Multi-Input/Multi-Output MIMO) networked control systems based on scalable control algorithms for uncertain time-varying nonlinear complex dynamic processes.

The major innovation of the proposal implies the elaboration of a new methodology for designing hierarchical adaptive self-organizing control systems to be applied to complex production processes, such as: plasma energy release, chemical and biological processes, casting in metallurgy, oil refinery, and so forth.

2 PHILOSOPHY OF HIERARCHICAL CONTROL

Industry and academia have investigated a wide range of decentralized control architectures ranging from hierarchical decomposition to a completely decentralized (heterarchical) approach where individual controllers are assigned to subsystems and may work independently or may share

data/information.

The main disadvantage of heterarchical approaches is that global optima cannot be guaranteed and predictions of the system's behaviour can only be made at the aggregate level. Hierarchical and heterarchical architectures lie at opposite ends of the distributed control architectures spectrum. The hierarchical approach is rigid and suffers from many of the shortcomings of the centralized approach, whereas it provides clear advantages in terms of overall system coordination alternatively. The heterarchical approach is flexible and fault-tolerant, but arguably difficult to coordinate.

Hierarchical control and supervision schemes have been widely studied as a possible solution for optimizing complex systems. A variety of schemes can be implemented in order to profit from the advantages of this architecture, and the applications run all the way from fault-tolerant aircraft control problems to servo systems supervision (Kwong et al., 1995).

Hierarchical control allows any available data from the low-level control system to be used at a higher level to characterize the system's current behaviour. Moreover, hierarchical control can be used to integrate extra information (in addition to that concerning the usual control-loop variables such as output, error, etc.) into the control decision-making process. In many situations a hierarchical approach is an advantageous option for process optimization, instead of sophisticated design and implementation of high-performance low-level controllers, because the hierarchical approach can compensate factors that are not taken into account in the design of low-level controllers (Berenji et al., 1991).

Thanks to its own structural essence, the hierarchical control scheme ensures flexibility and compatibility with other controllers that have already been installed. It has other strong points as well, such as the relatively low cost of investments in improving automation scheme performance, the possibility of exploiting already-installed low-level regulation systems, and the relatively low cost of measurement systems which makes hierarchical control a wise choice from economic and practical viewpoints.

The hierarchical methodology will cover three basic levels, namely:

I) physical control level composed of controlled process interacting with robust or classical controller through sensors and actuators;

II) adaptive level that implements scalable adaptation algorithms and enables the robust controller to satisfy a number of time-varying constraints;

III) knowledge-based self-organizing level which executes a set of self-learning, self-configuring, self-optimizing, and decision-making algorithms allowing possible dynamic changes in the system architecture aimed at optimal control and dynamic reconfiguration of the robust controller and making it easier to satisfy process-critical constraints, such as respond to the deadlines, saturations and so on.

Advanced algorithms designed for all control levels I, II, III and their interactions provide a *high degree of system flexibility, robustness, accuracy, reliability, dependability, and survivability* to deal with disruptive, uncertain, and unforeseen events *in automatic mode* (without human intervention).

The philosophy of the three-levels intelligent hierarchic control system proposed is a strategy of the project to be organized. Complex processes under control should be first of all thoroughly investigated, then classical or robust control systems are to be developed. After that, improvement of lower level should be done by means of levels II and III including decision making approach which may be performed by experts at the beginning of system design and not in automatic mode.

3 STATE-OF-THE-ART

The design of the main SISO and MIMO control loops should be based on, in a certain sense, classical approaches which have been used in modeling and practice and demonstrated efficacy in a number of applications. These approaches are based on design principles for which reliable dynamic processes models are created by means of First Principle Equations methodology (Khayrutdinov et al., 1993, Leonov et al., 2005), identification (Mitrishkin, 2004), linearization and subsequent reduction procedure. The resulting real world models have been used in system closed-loop to predict optimal control actions at each discrete timing interval to achieve the control goal taking into account input constraints (Mitrishkin and Korostelev, 2008). Decoupling control leads to the design of astatic multivariable systems (Leonov et al., 2005). A number of optimization approaches, specifically off-line Linear Quadratic techniques (Belyakov et al., 1999), H_∞ and μ synthesis

(Mitrishkin et al., 2003, Ariola and Pironti, 2008) as well as online automatic optimization of extremum search (Mitrishkin, 2004), gave acceptable results. Adaptive Kalman filter made it possible to estimate process parameters in real time and gave a chance to adapt the system by adaptive algorithm to time-varying process parameters (Mitrishkin and Kuznetsov, 1993). As this took place, an external disturbance was estimated and estimation value was used to compensate the disturbance itself (Mitrishkin, 2004).

Some results in the topics in relation to this proposal are the design and implementation of: intelligent hierarchical control and supervisory systems (Peres et al., 1999); control strategies by fuzzy, neural, neuro-fuzzy, and evolutionary neuro-fuzzy internal model control systems (Haber et al., 2004); rapid control prototyping for networked and embedded control (Haber et al., 2008); embedded intelligent control systems in open architectures (Haber and Alique, 2007), and networked control and intelligent monitoring for macro- and micro-scale manufacturing processes (Haber et al., 2008).

The synthesis of intelligent hierarchical control system (Andrikov and Konykov, 2004) was applied to improve car braking controllability by H_∞ control theory.

4 STATEMENTS OF CONTROL PROBLEMS

A number of new important complex control problems have to be studied, discussed and formulated to achieve control goals of acceptable trade-off between robust stability and performance of feedback systems. The problem statements concern the stabilization and tracking process output signals, optimal distribution of process parameters in space in the presence of non-modeled process dynamics, unobserved disturbances, nonlinearities, in particular saturations, wideband insufficiently known noise in output signals, non-minimum-phase dynamics, and time-varying parameters. To solve these control problems a set of approaches from linear and nonlinear control theory will be explored and developed to achieve scalable H_∞ robust, decoupling, model predictive, adaptive, hierarchy, cascade, soft-computing based control (e.g., neuro-fuzzy control systems), and facilitate decision-making in new appropriate combinations within continuous and discrete time of the three-level hierarchic control system. Scalable control algorithms mean that the algorithms may be generalized to any numbers of controlled plant inputs, outputs, and space states.

5 PRACTICAL APPLICATIONS

In order to validate the suggested approaches plasma energy release case study is planned to be investigated. Control methodologies will be applied to plasma vertically unstable position, shape, and current in the presence of voltages and *current saturations* in poloidal magnetic coils in ITER (International Thermonuclear Experimental Reactor, www.iter.org). Multivariable robust controller design (level I) *with adaptation* on the level II is proposed to be done for the *whole plasma discharge* of plasma current ramp-up, ramp-down, and at quasi stationary stages. It is planned to be applied for ITER reference scenario No 2 with plasma current on flat-top of 15 MA and for reversed share scenario No 4 of plasma current of 9 MA. Mathematical modeling of control systems to be developed on plasma-physics code DINA (Khayrutdinov et al., 1993) is assumed to be fulfilled in tracking and stabilizing modes at disturbances of minor disruption type.

The project control methodologies are planned to be applied to solve *plasma kinetic control problem* as well. Plasma kinetic control means creation and maintenance of *optimal* plasma current, temperature, and density profiles by means of additional heating sources. Such regimes are necessary for stationary operation of tokamak-reactors. Development of kinetic plasma models of plasma current, temperature, and density profiles and their identification are supposed to be created. Then design and modeling of plasma profiles control systems are assumed to be performed.

The final issue of this activity is integration of plasma magnetic and kinetic control systems.

In the process of hierarchical structure control systems design the synthesis, analysis, and numerical modeling approaches are proposed to be performed in MATLAB/SIMULINK environment.

The ITER functionality is planned to be performed by means of CODAC software specifically Control, Data Access, and Communications (www.iter.org) via which one can install hierarchical control scheme proposed.

6 CONCLUSIONS

The concept of three levels hierarchic control system was presented and discussed namely: architectural details, state-of-the-art, statement of control problems, and practical applications.

The project will result in the creation of new process models, procedures of their identification and reduction, efficient, robust, predictable, and safe ICT control methodologies, scalable control algorithms, and high-performance controllers with reconfigurable architecture for the problem oriented hierarchical systems under consideration. Scientific, engineering, and industrial results will be accumulated in the data and knowledge bases with accurate classification, qualitative and quantitative assessment, and generalization.

REFERENCES

Albus J. S., 1991. Outline for a theory of intelligence, *IEEE Trans. Syst., Man, Cybern. A* Vol. 21 (3), pp. 473-508.

Andrikov D., Konykov V., 2004. Robust H_∞ – optimal controller for car with ABS in emergency situation in slip mode. *Herald of BMSU, Instrument engineering series.* Vol. 57, No. 4, pp. 44–57 (in Russian).

Ariola M., Pironty A., 2008. *Magnetic Control of Tokamak Plasmas*. Springer-Verlag.

Belyakov V., Kavin A., Kharitonov V., Misenov B., Mitrishkin Y. et al. 1999. Linear Quadratic Gaussian Controller Design for Plasma Current, Position and Shape Control System in ITER, *Fusion Engineering and Design*, Vol. 45, pp. 55-64.

Berenji H. R., Chen Y.-Y., Lee C.-C., Yang J.-S., Murugesan S., 1991. A hierarchical approach to designing approximate reasoning-based controllers for dynamic physical systems, in *Uncertainty in Artificial Intelligence* Vol. 6, pp. 331-343, 1991.

European Commission C (2008)6827, 17 November 2008. Work Programme 2009. Cooperation, Theme 3. *ICT – Information and Communication Technologies*, p. 38.

Haber R. E., Alique J. R., 2004. Nonlinear internal model control using neural networks: an application for machining processes. *Neural Computing & Applications,* vol. 13, pp. 47-55.

Haber R.E., Villena P., Haber-Haber R., Alique J.R., 2008. Fast design and implementation of intelligent controllers. *DYNA*, vol. 83 (8), pp. 127-134.

Haber R. E., Alique J. R., 2007. Fuzzy logic-based torque control system for milling process optimization. *IEEE Trans. on Systems Man and Cybernetics. Part C-Applications and Reviews,* vol. 37, pp. 941-950.

Haber R. E., Martin D., Haber-Haber R., Alique A., 2008. Networked fuzzy control system for a high-performance drilling process. *Journal of Manufacturing Science and Engineering-Trans. of the ASME*, vol. 130, pp. 68-75.

Khayrutdinov R.R., Lukash V.E., 1993. Studies of Plasma Equilibrium and Transport in a Tokamak Fusion Device with the Inverse-Variable Technique. *Journal Comp. Physics*, Vol. 109, pp. 193–201.

Kwong W. A., Passino K.M., Laukonen E.G., Yurkovich S., 1995. Expert supervision of fuzzy learning systems for fault tolerant aircraft control, *Proc. of IEEE* Vol. 83 (3), pp. 466-483.

Leonov V., Mitrishkin Y., Zhogolev V., 2005. Simulation of Burning ITER Plasma in Multi-Variable Kinetic Control System. *Proc. of 32nd Plasma Physics Conf. of European Physics Society*, Tarragona, Spain, ID P5.078.

Mitrishkin Y., Kuznetsov E., 1993. Estimation of Parameters of Stabilized Plasma. *Plasma Devices and Operations*, No. 3, Vol. 2, pp. 277-286.

Mitrishkin Y., Kurachi K., Kimura H., 2003. Plasma multivariable robust control system design and simulation for a thermonuclear tokamak-reactor, *International Journal of Control*, Vol. 76, No. 13, pp. 1358-1374.

Mitrishkin, Y., 2004. Comprehensive Design and Implementation of Plasma Adaptive Self-Oscillations and Robust Control Systems in Thermonuclear Installations. *Proc. of 8th World Multi-Conference on Systemics, Cybernetics and Informatics*, Orlando, FL, USA, Vol. XV, pp. 247-252.

Mitrishkin Y., Korostelev A., 2008. System with Predictive Model for Plasma Shape and Current Control in Tokamak. *Control Sciences*, No.5, pp. 22-34 (in Russian).

Peres C. R., Haber R. E., Haber R. H., Alique A., Ros S., 1999. Fuzzy model and hierarchical fuzzy control integration: an approach for milling process optimization. *Computers in Industry*, vol. 39, pp. 199-207.

A NOVEL POTENTIAL FIELD ALGORITHM AND AN INTELLIGENT MULTI-CLASSIFIER FOR THE AUTOMATED CONTROL AND GUIDANCE SYSTEM (ACOS)

Thomas Statheros, Gareth Howells, Pierre Lorrentz
Department of Electronics,University of Kent, Canterbury, Kent, U.K.
{ts40, W.G.J.Howells, pl38}@kent.ac.uk

Klaus McDonald-Maier
School of Computer Science and Electronic Engineering, University of Essex, Colchester, U.K.
kdm@essex.ac.uk

Keywords: Autonomous Intelligent Guidance, Potential Field Algorithms, Weightless Neural Systems.

Abstract: The ACOS project seeks to improve and develop novel robot guidance and control systems integrating Novel Potential Field autonomous navigation techniques, multi-classifier design with direct hardware implementation. The project development brings together a number of complementary technologies to form an overall enhanced system. The work is aimed at guidance and collision avoidance control systems for applications in air, land and water based vehicles for passengers and freight. Specifically, the paper addresses the generic nature of the previously presented novel Potential Field Algorithm based on the combination of the associated rule based mathematical algorithm and the concept of potential field. The generic nature of the algorithm allows it to be efficient, not only when applied to multi-autonomous robots, but also when applied to collision avoidance between a single autonomous agent and an obstacle displaying random velocity. In addition, the mathematical complexity, which is inherent when a large number of autonomous vehicles and dynamic obstacles are present, is reduced via the incorporation of an intelligent weightless multi-classifier system which is also presented.

1 INTRODUCTION

This paper presents additional novel algorithms, methods and technologies adapted by the ACOS automated guidance system (Statheros, 2006) for collision free autonomous navigation, not only in a single autonomous manner, as initially presented in (Statheros et. al., 2006), but also for multi-autonomous vehicles in the presence of independent dynamic obstacles.. The technologies employed fall into three major categories: Novel Potential Field autonomous navigation techniques, multi-classifier design and direct hardware implementation. This paper presents an overview, further development and ideas regarding the integration of these technologies within the ACOS system. The paper presents the novel features of the Potential Field methodology described in (Statheros 2007), and also the new concept of Trajectory Equilibrium State (TES) between a potential field autonomous vehicle and a dynamic obstacle. In addition, we propose the combination of the multi-classifier with the novel potential field algorithm in a new hybrid navigation system. This is followed by a description of the multi-classifier framework employed by ACOS which utilises weightless neural network technology allowing a rapid adaptable learning environment and facilitating efficient direct hardware implementation. The multi-classifier additionally possesses the desirable properties of 1) a capacity to implicitly adapt to the relative discriminant abilities of its component classifiers and 2) be able to accept both absolute and probability based classifications from its component classifiers.

2 NOVEL POTENTIAL FIELD METHOD FOR MULTI-AUTONOMOUS VEHICLE NAVIGATION

A major part of ACOS work for autonomous navigation is based on novel potential field algorithmic methodology improving both single and multi-autonomous vehicle navigation (Statheros 2007). The generic concept of "artificial potential fields" originates from (Khatib, 1985). This study introduces the potential field method (PFM) for real-time obstacle avoidance for both manipulators and mobile robots. In later years PFM quickly gained popularity for autonomous vehicle navigation because of its elegance and simplicity. A Widely used PFM for mobile robot real-time obstacle avoidance is termed Virtual Force Field (VFF) (Borenstein, 1989, 1990). The VFF method has also been utilised in complex hybrid systems for air, land and water based autonomous navigation. A number of VFF algorithms specialised in water based navigation are briefly explained in (Statheros, 2008).

However, Artificial potential field based algorithms experience local minima traps, which cause autonomous vehicle's trajectory deadlocks and/or oscillations (Koren 1991). This problem can be resolved by PFM in integration with intelligent methods and/or mathematical navigational algorithms.

In recent years potential field algorithms have also gained popularity in the field of multi-autonomous navigation (Pradhan 2006, Masoud 2007). In (Statheros 2007) a novel multi-autonomous navigation algorithm enables a simple VFF algorithm to navigate local multi-autonomous independent vehicles exceptionally efficient in terms of trajectory length, trajectory smoothness and time of arrival. This approach uses a novel rule-based mathematical algorithm and the newly defined concept of trajectory equilibrium state (TES).

2.1 VFF Trajectory Equilibrium State

In a multi-mobile robot environment where the robots are guided by the VFF method, in which the virtual repulsive force is described in (Statheros 2007), we can observe the Trajectory Equilibrium State (TES) as shown in Figure 1. Here, we observe that the robot trajectories cross at point C to reach their target destinations in straight line trajectories. However, with VFF, the trajectory diversion leads to autonomous navigational deadlock and both robots stop at points D and E without reaching their target destinations T1 and T2. We can define the distance DE as $D_{Saturation}$, the minimum distance they may have between them. The robots will only stop without reaching their target destination in Absolute TES. Where equation (2) is not fully satisfied but equation (1) is satisfied, we define the state as Close TES.

$$D_{Saturation} \leq D < D_{Efficiency} \quad (1)$$

$$AC = BC \text{ and } V_1 = V_2 \quad (2)$$

In equation 1, $D_{Efficiency}$ is the minimum distance between the two robots so the non-linear effect of the equation 1 is not apparent. Where V1 is the speed of mobile robot 1 and V2 is the speed of mobile robot 2.

As stated above, the TES causes trajectory inefficiencies such as long and curved power consuming trajectories for all the guided robots. In the most extreme case, absolute TES, both robots divert from their target destination and the distance between them decreases to the point where the resultant force vectors are equal to zero. The Absolute TES has been identified utilizing two mobile robots in (Statheros 2007).

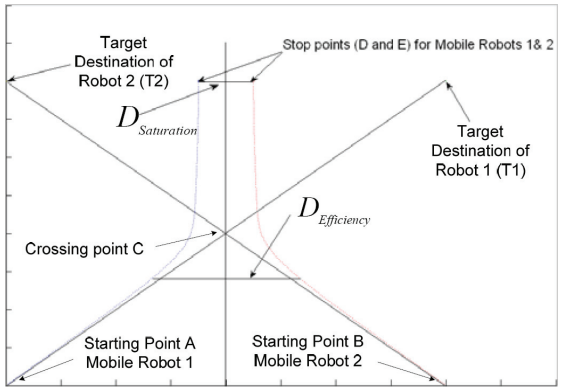


Figure 1: Two mobile robots at Absolute Trajectory Equilibrium State (TES).

2.2 TES Detection and Avoidance

The TES detection and avoidance algorithm predicts and prevents Absolute and Close TES. This algorithm maintains close to straight line efficient trajectories for the robots in cases of possible collision by adjusting separately their speeds. The performance of this algorithm is demonstrated in Figure 2.

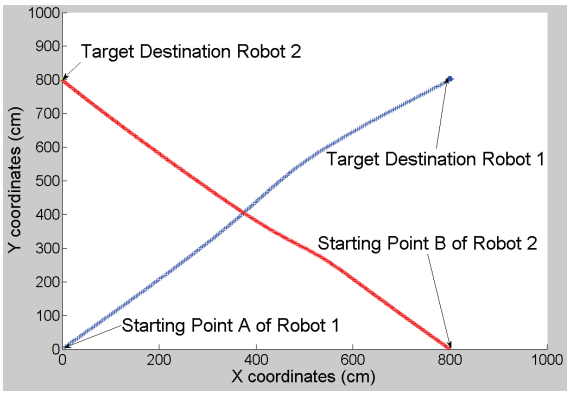


Figure 2: Two mobile robots close-optimum trajectories due to TES Detection and Avoidance algorithm in case of Absolute TES.

The above has introduced the concept of guiding independently multi-autonomous robots or vehicles with identical algorithmic principle with exceptional efficiency. However, in this paper, we have identified that the above algorithm is more generic in nature, as it may also be applied to dynamic obstacles. For example, in Figure 3, a collision scenario is presented between a dynamic obstacle and a standard potential field guided robot. In this case, we can consider a new concept of TES between a potential field robot and a dynamic obstacle. This TES forces the potential field guided robot to divert from its target destination and follow the inefficient trajectory shown in figure 3. The TES detection and avoidance algorithm can also be applied in this case. The algorithm incorporates a velocity variation of the autonomous guided robot based on the potential field algorithm dynamics. The effectiveness of the algorithm is displayed in Figure 4, where the autonomous vehicle follows a near optimum straight line trajectory.

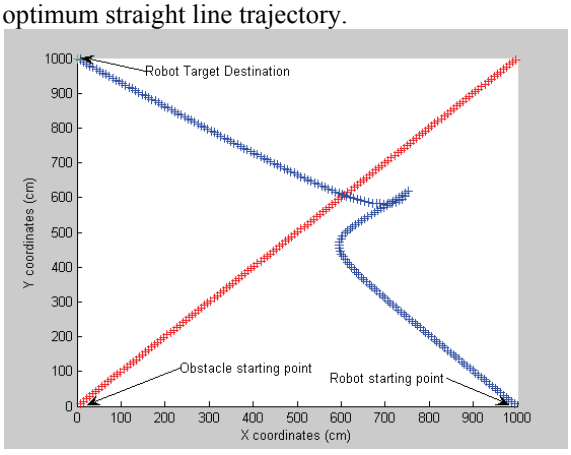


Figure 3: Standard Potential Field robot with dynamic obstacle.

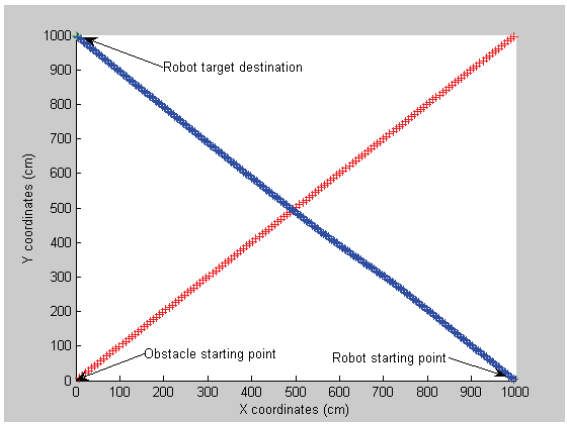


Figure 4: The effect of TES Detection and Avoidance algorithm when a Potential Field robot is in TES with a dynamic obstacle.

The processing requirements of the above algorithm increase in a presence of a large number of autonomous vehicles and/or dynamic obstacles. We can reduce its processing load by focusing the algorithm onto a group of similarly behaving dynamic vehicles and/or obstacles that are recognised by an intelligent multi-classifier, which we present in the next section. This is possible due to the patterns of location, direction, speed and potential field algorithm dynamics, which are generated from the autonomous vehicles and/or dynamic obstacle in the same local navigation environment).

3 THE INTELLIGENT FAST-LEARNING MULTI-CLASSIFIER SYSTEM

Modern intelligent Robotic Guidance systems are being employed in practical application domains where the required performance level often exceeds that achievable from a single guidance paradigm typically because the complexity of the problem is such that too many potential outcomes are present, equivalent to the number of pattern classes when the system is viewed as a pattern recognition problem. To address this issue, current systems often concurrently employ a number of distinct classifiers, where the component classifiers are trained on a subset of situation which the robotic system may encounter in practice. Therefore, the component classifiers will possess the ability to distinguish well between certain situations but will be unable to offer the same distinguishing pattern classification performance over the entire range of scenarios

specific to the problem domain because they are unaware of all possible situations. In such circumstances, engineering a solution to a practical problem is reduced to a selection process of available classifiers where the combination of the classifiers chosen is able to distinguish the entire set of pattern classes present within the problem domain. A combiner classifier is required in addition which is trained on the outputs of the component, or base, classifiers and makes an overall decision.

The ACOS system utilises an intelligent multi-classifier combiner system which is able to automatically assimilate outputs of component system classifiers which are inaccurate due to their restricted training knowledge and produce a single classification for a given classification instance The system possesses the following significant properties:-

- All base classifiers and the combiner classifier follow a generic architecture based on the Probabilistic Convergent Network (PCN) (Howells 2000, Lorrentz 2007).
- The significance of the classification decision of a given classifier is varied according to the likely pattern classes under consideration. Therefore, a classifier which possesses good knowledge of the scenario in question is able to provide a strong weighted decision which is utilised by the combiner network. Conversely, when an unfamiliar scenario is encountered, a low weighted incorrect decision is produced due to the unfamiliarity of the classifier with the true scenario.
- The multi-classifier system possesses fast learning properties so that the significance of class distinguishing properties are immediately accepted by the system
- The system is problem domain independent and may be adapted to a large number of automated navigation based scenarios.
- The system uses simple logic operations to guide its decision making process and it is thus suitable for fast direct hardware based implementation

As stated, the proposed technique employs a type of weightless artificial neural system known as the Probabilistic Convergent Network (PCN) to assimilate the classification potential of each of the component classifiers employed in a given situation. The PCN network architecture (Howells 2000, Lorrentz 2007). is designed to provide an extended recognition information base to the user whilst retaining the training and performance potential achieved with previous Weightless architectures (Austin 1998). An example PCN architecture is illustrated in Figure 5.

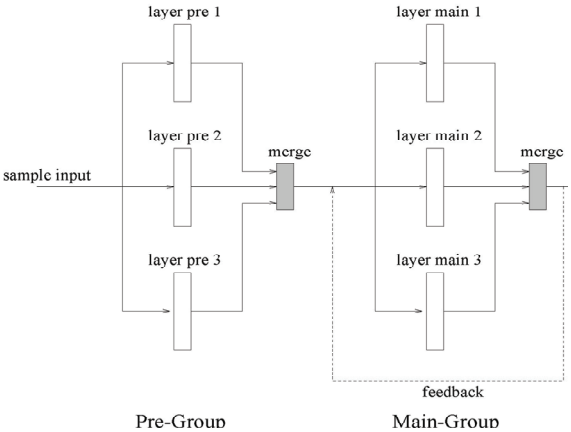


Figure 5: PCN Network Architecture.

The following are significant points regarding the architecture:-

- The neurons comprising the network are arranged in $x \times y$ matrices or layers where x and y are the dimensions of the input sensor data under consideration.
- Each element within the sensor data is therefore associated with a corresponding neuron within each layer.
- The layers comprising the network are arranged in two groups, termed the *Pre* group and the *Main* group. A *Merge* layer exists after each group whose function is to combine the outputs of the constituent layers of the group. The connectivity of the neurons comprising a *Merge* layer is equal to the number of layers within the group to which it pertains.
- The merged output of the *Main* Group is fed back, unmodified, to the inputs of each layer comprising the group.
- The number of layers within each group may be varied depending on the recognition performance required from the network.
- The constituent layers of a group differ in the selection of sensor data elements attached to the inputs of their constituent neurons (termed the *connectivity pattern*).

- Neurons within a given layer possess the same connectivity pattern relative to their position within the matrix.

The PCN architecture utilises highly efficient training and recognition algorithms which are detailed in (Lorrentz 2007). These allow the network to produce weighted decisions on their output giving a confidence level associated with the decision. Specifically:-

- Symbols within the PCN architecture are taken from an extended *compound* set.
- A given symbol is designed to contain a component for each of the possible pattern classes on which the network has been trained.
- Each component itself is constructed from a pre-determined number of sub-symbols. This number represents the number of *divisions* available for each pattern class where each divisional symbol represents a probability approximation that the given sample pattern belongs to the given pattern class.

The neurons comprising the network differ between the *Pre* and *Main* groups. The *Pre* group neurons take their inputs from the binary sensor values comprising the network input data. The contents of the memory locations of the neurons are taken from the extended compound set of symbols described above. The main group neurons take their inputs and memory contents from the compound set of symbols.

Due to the weightless nature of PCN it lends itself to straightforward hardware implementation that requires mainly standard memory to realise the network structure and some limited arithmetic resources. An enhanced version of the PCN architecture has been prototyped and forms a hardware fabric the for the systems implementation (Lorrentz 2008).

The ACOS system consists of several base PCN base classifiers based on separate scenarios which a robot may encounter. It is infeasible to train a single PCN classifier with a large number of scenarios due to the exponential increase in memory required as each neuron memory will increase in size for each new scenario. The PCN architecture naturally lends itself to employment as an intelligent multi-classifier however. To achieve this end, the output classifications of the selection of base classifiers employed, form the input to a given combiner PCN classifier. The outputs of the combiner PCN will then represent a weighed classification for the problem at hand based on the combined wisdom of the component classifiers as illustrated in Figure 6.

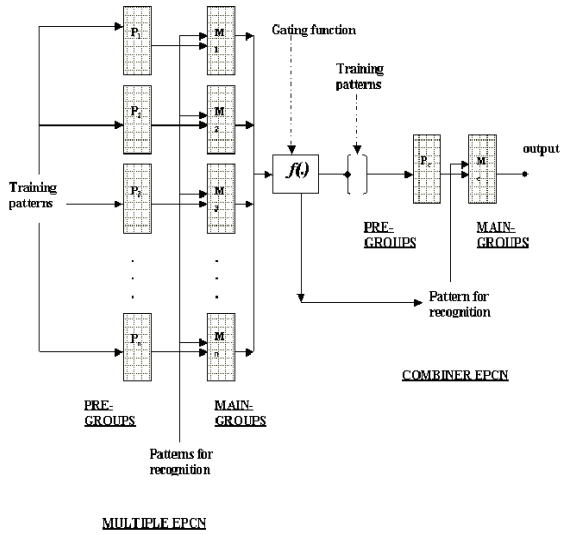


Figure 6: Schematic of the PCN based Multi-Classifier.

As stated, in order to employ the PCN architecture as a basis for a multi-classifier system, it is necessary to combine the outputs of the component classifiers to form a single input which may be considered as a classification image for the particular problem in question. The general strategy requires the following steps to be taken:-

- Outputs of component classifiers are interpreted as binary numbers, either indicting a single preferred pattern class or representing a combination of classed with associated probabilities.
- The combiner PCN overloads the meanings of the outputs of the component classifier in order to address the memory scale issue associated with the requirement that it be able to distinguish between a large number of component decisions. So, for example, the meaning of class decision 1 for base classifier 1 will differ from the same output for classifier 2. However, the combiner PCN sees a compound input pattern which essentially represents a compressed representation of all possible decision scenarios with associated weightings and is able to efficiently reach a conclusion.
- Suitable training examples must be compiled which will allow the PCN system to distinguish between the various scenarios. To this effect it is a supervised learning environment.

Examples of classifications may now be

presented to the PCN architecture according to the training algorithm in (Howells 2000, Lorrentz 2007). The system effectively relies of the fact that if a base classifier encounters a situation with which it is familiar (i.e. it has encountered in training), it will produce a decision with high confidence. Conversely, if a base classifier encounters a scenario with which it is not familiar, it will produce a classification from one of the scenarios which it is familiar but with low confidence. i.e. it will produce an erroneous but low weighted result. The combiner PCN is able to sift these decisions and produce the desired decisions based on their confidence rating.

4 CONCLUSIONS

The ACOS project has been successful in producing an integrated, automated, robotic guidance system which is highly flexible and capable of fast autonomous learning. It has achieved its primary aim of providing state-of-the-art knowledge on autonomous navigation techniques and technologies as well as a novel autonomous navigation techniques architecture which constitutes design and implementation suitable for industrial exploitation.

ACKNOWLEDGEMENTS

This research is supported by the European Union ERDF Interreg IIIa scheme under the ACOS Grant.

REFERENCES

Statheros, T., Howells, G, McDonald-Maier, K.D, 2007 Trajectory equilibrium state detection and avoidance algorithm for multi-autonomous potential field mobile robots. *Electronics Letters, 43(15): p. 799-801.*

Khatib, O. 1985 *Real-time obstacle avoidance for manipulators and mobile robots.* 2: p. 500.

Borenstein, J., Koren, Y. 1989. Real-time obstacle avoidance for fast mobile robots. *Systems, Man and Cybernetics, IEEE Transactions on, 1989. 19(5): p. 1179.*

Borenstein, J., Koren, Y. 1989, Real-time obstacle avoidance for fast mobile robots in cluttered environments. *Proceedings., IEEE International Conference on, 1990: p. 572 - 577.*

Statheros, T., Howells, G, McDonald-Maier, K.D, 2008 Autonomous ship collision avoidance navigation concepts, technologies and techniques. *Journal Of Navigation.. 61: p. 129-142.*

Koren, Y., Borenstein, J., 1989. Potential field methods and their inherent limitations for mobile robot navigation. p. 1398.

Pradhan S.K. et al 2006, Potential field method to navigate several mobile robots. *Applied Intelligence. 25(3): p. 321-333.*

Masoud, A.A., 2007. Decentralized self-organizing potential field-based control for individually motivated mobile agents in a cluttered environment: A vector-harmonic potential field approach. *IEEE Transactions On Systems Man And Cybernetics Part A-Systems And Humans,. 37(3): p. 372-390.*

Statheros, T et. al., 2006. Automated Control and Guidance System (ACOS): An overview *Sixth International Conference on Recent Advances in Soft Computing, Canterbury, UK,*

Howells, G., Fairhurst, M.C., Rahman, F. 2000. An exploration of a new paradigm for weightless RAM-based neural networks, *Connection Science, Vol. 12, No. 1 pp. 65-90.*

Lorrentz, P. Howells, G., McDonald-Maier, K.D., 2008: An FPGA based adaptive weightless Neural Network Hardware, *IEEE, NASA/ESA Conference on Adaptive Hardware and Systems 2008, AHS-2008*, Noordwijk, The Netherlands.

Austin, J. (ed.) 1998 *'RAM-based Neural Networks'* World Scientific ISBN 981-02-3253-5

Lorrentz, P. Howells, G., McDonald-Maier, K.D., 2007. Design and Analysis of a novel weightless artificial neural based Multi-Classifier, International Conference of Computational Intelligence and Intelligent Systems (ICCIIS 2007), *part of World Congress on Engineering 2007 (WCE 2007),* London, UK.

A PRACTICAL APPROACH FOR COMBINATORIAL FUZZY LOGIC CONTROL DESIGN

Arturo V. Téllez, Luis A. V. Villa, Herón L. Molina, Oscar N. Camacho and Romeo P. Urbieta
Centro de Investigación en Computación, Instituto Politécnico Nacional
Juan de Dios Batiz Ave. s/n, Nueva Industrial Vallejo, Mexico City, Mexico
vitrion@hotmail.com, lvilla@cic.ipn.mx, oscarc@cic.ipn.mx, rurbieta@cic.ipn.mx, hmolina@cic.ipn.mx

Keywords: Fuzzy Logic Controller, Fuzzy Sets Adaptation, Combinatorial Design, FPGA, Fast Prototyping, Hardware.

Abstract: This paper presents the architecture development of a Fuzzy Logic Controller (FLC), using combinatorial design implemented on a Field Programmable Gate Array (FPGA). This architecture is based on combinatorial basic modules that enable to increase and improve the entire system performance, by means of replication technique, which is widely used in computer architecture, and help to fit the particular application needs. Recent FPGA technology let us use fast combinatorial circuits for complex designs with parallelism for increasing the FLC performance and it is possible to take it up again as a practical way to build FLC for any process, approaching the fast prototyping advantages and easing the scaling to increase the control accuracy.

1 INTRODUCTION

A FLC can be implemented in software easily and executed in a microprocessor, a microcontroller, or a general purpose computer. Though software- based FLC are cheaper and flexible, there are some difficulties when control systems require high data processing. The use of FPGA has been profitable when talking about versatility to make any digital design by means of costs and design time. In principle, the implementation of FLC is not based on the mathematic model of the plant, but this kind of system is very effective to control a process where the transfer function is not known, instead the control action is based on the extern influence and simple decisions based on a knowledge base acquired with experience, the same way a human would do it, exploiting the heuristic ability. There have been so many FLC implementations since the first hardware one appeared (Togai, 1986), which used complex designs with sequential circuits because of the high hardware resource and delay time costs about combinatorial design. A large quantity of FLC architectures, derived from Computing Architecture. These architectures are classified by its processing way. There are sequential, combinatorial (Manzoul, 1992), parallel, pipelined and mixed models. Some designers prefer to implement these operations to calculate a parameter of the FLC every time it is necessary (Gaona, 2003); this technique is called Runtime Computation (RTC). But some designs use extern elements like memories, sometimes called Look Up Tables (LUT), to calculate FLC parameters by anticipation; this another technique is called Look Up Computation (LUC) and represents a good way to improve the timing (Vasantha, 2005; Singh, 2003; Deliparaschos, 2005). It is a dare to play with these architectures and techniques to make a balanced FLC design, by which it is necessary to change the way of designing algorithms to describe a FLC.

This paper shows a practical approach of FLC combinatorial architecture in order to make simple construction modules and easy upgrading using a reprogrammable device, FPGA.

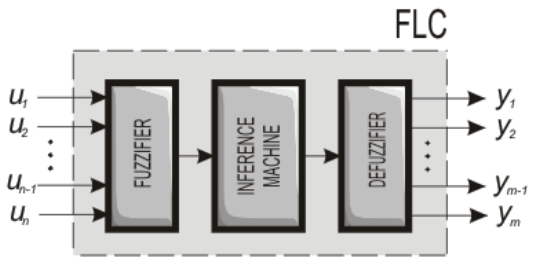


Figure 1: Fuzzy Logic Controller.

2 SYSTEM DESCRIPTION

Assume u_i as the inputs to the FLC and y_j as the outputs. Figure 1 shows a FLC which consists of three basic stages: Fuzzification, Inference Machine and Defuzzification. The *Fuzzification* stage consists of fuzzy sets. Each fuzzy set converts every crisp input into several fuzzy values or membership values. The *Inference Machine* contains the behaviour of the FLC and it is built with MIN-MAX modules. These rules have simple inferences of the type IF- THEN. Also, the *Defuzzification* stage converts these inferred values onto crisp values, by means of statistical calculations, which represent the control action over the actuator. The next steps are required for build a FLC (Tellez, 2008):

1. Establish whatever the designer want to control and which variables will be related to get it.
2. Define the number of inputs and outputs of the FLC based on the last step.
3. Define the number of membership functions or fuzzy sets for each input and output based on the last step and define their shape based on the process characteristics and operation range of the FLC (discourse universe).
4. Set the FLC configuration by means of the fuzzy inference rules according to the wished operation and based on the expert knowledge about the process.
5. Build the fuzzifier with simple membership functions simply by replication (trapezoidal, triangular, S, Z).
6. Build the inference machine based on step 4, by means of MIN- MAX modules using the building steps shown in section 2.2.
7. Once inference machine is ready, build the defuzzification stage by means of multiplication and division modules using parallelism.
8. Finally, FLC can be implemented on FPGA.

For the FLC implementation it was used VHDL, Xilinx ISE 6.3i, Mentor Graphics Modelsim Xilinx Edition III 6.0a. It is used Xilinx Spartan 3 XC3S200–5FT256 FPGA Starter Kit. In order to verify the FLC performance, it was necessary to make a simulation using the Fuzzy Toolbox of MATLAB and build a control system with SIMULINK.

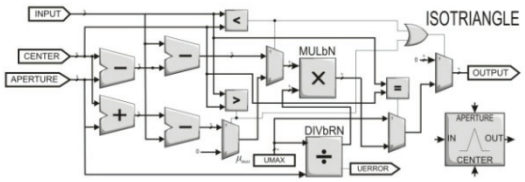


Figure 2: Isosceles triangular membership function shape.

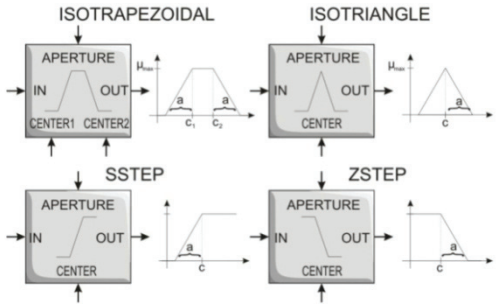


Figure 3: Several hardware suitable membership functions.

2.1 Fuzzification

The fuzzification stage comprises a set of fuzzifiers attached to every input variable; each one parallel from the others and their performance does not depend on the others either. We assume that all membership functions shape will be triangular, trapezoidal, S and Z, because they are the easiest to implement in hardware as shown in the Figure 2. These modules convert a crisp digital value into a membership digital value, according to two parameters: the CENTER and the APERTURE. These two parameters of the membership functions accomplish the RTC technique in order to make the online adaptation and the FLC tuning.

These functions may have several shapes as shown in Figure 3 and the interconnection seems like it follows in Figure 4. Next section describes the inference machine construction according to a set of steps using Mamdani operation.

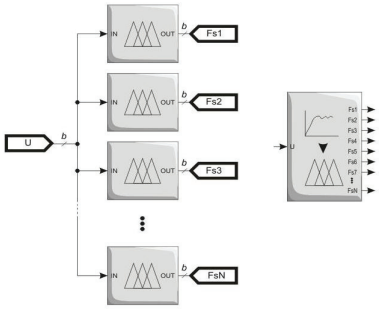


Figure 4: Fuzzifier.

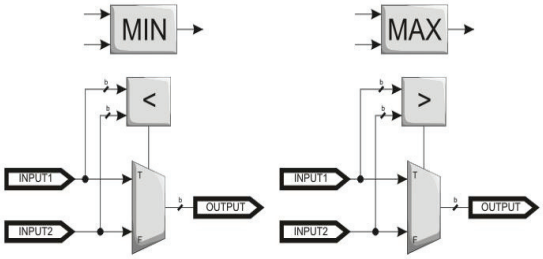


Figure 5: MIN- MAX modules.

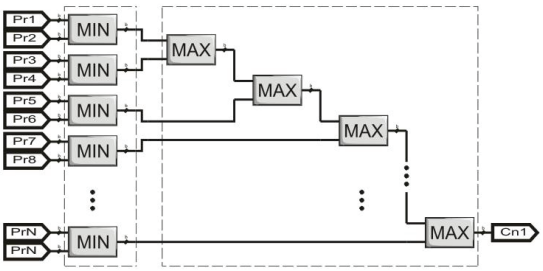


Figure 6: Inference machine stage construction.

2.2 Inference Machine

Let us define a *premise* as the input data involved with the control, it means that an involved input will be considered to decide which control action will be taken. A *consequence* is a result of the inference, the output data of inference machine, it means the decision that FLC will take based on the premises. A fuzzy rule set is the FLC configuration of the simple form:

IF $premise$ 1 **AND** $premise$ 2 **AND**, … , **AND** $premise\ n$
THEN $conseq$ 1 **AND** $conseq$ 2 **AND**, … , **AND** $conseq\ m$

A Mamdani inference machine consists of MAX-MIN (Figure 5) modules interconnected according to the fuzzy rule set (Patyra, 1996). A MAX-MIN structure of an inference machine has MIN modules in parallel. Unlike the MAX modules are in cascade, as shown in Figure 6.

2.3 Defuzzification

This stage obtains a crisp output by means of output fuzzy sets, sometimes called *Centroid* method. The calculation of the centroid is made using the membership values $\mu_i(u_1, u_2, \dots, u_n)$, obtained from the inference engine, and the output fuzzy set centers b_i^q. It is often considered as *singleton* membership function, because of its computational simplicity and because this statistical calculation is independent of the output fuzzy set shapes.

$$y_q^{crisp} = \frac{\sum_{i=1}^{R} b_i^q \mu_i(u_1, u_2, \dots, u_n)}{\sum_{i=1}^{R} \mu_i(u_1, u_2, \dots, u_n)} \quad (1)$$

This defuzzifier needs a division calculation, as seen in the Equation 1, which results computationally expensive when trying to divide $2b$ bits multiplication result numbers, which is not practical neither cheap computationally. In order to avoid the $b \times b$ multiplication before the division, so part of Equation 1 was implemented this way:

$$\sigma_i = \frac{b_i^q}{\sum_{i=1}^{R} \mu_i(u_1, u_2, \dots, u_n)} \quad (2)$$

Then, the result σ_i (Equation 2) is multiplied by every membership value obtained from the inference machine. To get this, it was needed to implement a combinatorial non-restoring division (Oberman, 1997) modified to obtain a fixed point 2b bits quotient, because $\sigma_i < 1$, as shown in Figure 7.

3 IMPLEMENTATION AND VERIFICATION

As example of application, a FLC for a DC servo is implemented, as mentioned above, in order to verify the correct performance of the FLC.

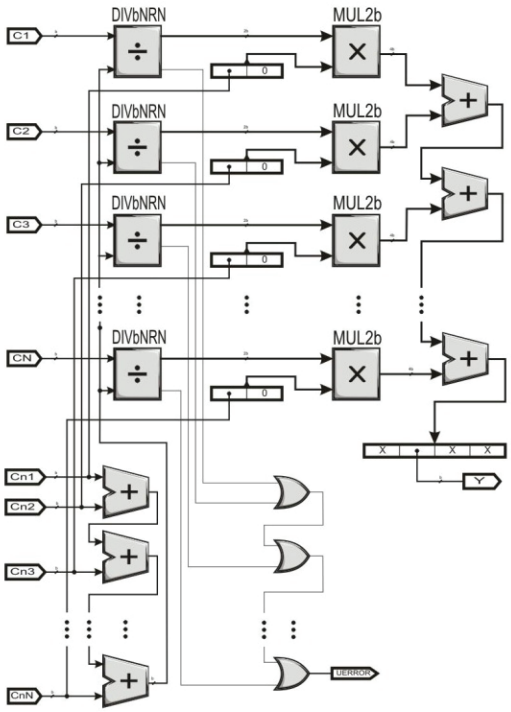


Figure 7: Defuzzification stage.

The control system was built in MATLAB Fuzzy Toolbox first, creating a fuzzy inference system by software (FIS). As an example, suppose that we want to implement a 2×1 fuzzy system for a DC servo, which uses nine rules because it has three fuzzy sets per input (position error eP: NE, ZE, PE. position error change velocity cP: NC, ZC, PC) and output (voltage <volts>- V: NV, ZV, PV), shown in the Figure 7, which are the following:

```
IF eP  is NE AND cP  is NC THEN V  is NV
IF eP  is NE AND cP  is ZC THEN V  is NV
IF eP  is NE AND cP  is PC THEN V  is NV
IF eP  is ZE AND cP  is NC THEN V  is NV
IF eP  is ZE AND cP  is ZC THEN V  is ZV
IF eP  is ZE AND cP  is PC THEN V  is PV
IF eP  is PE AND cP  is NC THEN V  is PV
IF eP  is PE AND cP  is ZC THEN V  is PV
IF eP  is PE AND cP  is PC THEN V  is PV
```

Then, it was provided a test bench which consists of 25 values and describes several input situations but due to space it is not explained in this paper. Also, FLC tuning was made changing the membership function parameters of inputs and outputs. Table 1 shows all timing and resources in FPGA used for every implemented module built for DC servo FLC example. DC servo FLC needs 84 ns to make a single inference. Then, its processing data rate is 11.9 MFLIPS.

Table 1: FPGA timing and resource results obtained for DC servo control.

Algorithm	Delay (ns)	LUT
16 bits non-restoring division	48.50	644
Modified 8 bits non-restoring division	28.83	208
8 bits restoring division	28.84	124
8 bits multiplication	13.17	36
Isosceles triangle MF	36.70 14.51	251
S-step MF	36.70	249
Z-step MF	36.70	251
Fuzzifier	37.42	755
Defuzzifier	41.49	677
Mamdani inference machine	19.32	242
MIN-MAX operations	9.36	16
FLC	84.01	2689

4 CONCLUSIONS

FLC architecture was designed using RTC combinatorial arithmetic modules. In order to get this, it was supplied to designer a practical approach for FLC design, using a study case (DC servo). Those developed modules were implemented in FPGA and it was possible to verify the FLC performance compared with the FIS simulated with MATLAB. We proved that this architecture has the capability of grow modularly. This modularity may be approached using a FIS to VHD language interpreter that simply generates the proper HDL program, using the basic modules presented in this paper, regardless the used technology, based on the MATLAB `*.fis` configuration file.

ACKNOWLEDGEMENTS

Research supported by the Instituto de Ciencia y Tecnología ICyTDF funding (award No. PICCT08-22) and by matching funding by IPN (award No. SIP/DF/2007/143).

REFERENCES

Téllez, A., 2008. *Fuzzy Logic Controller Architecture using Combinatorial Logic*, Instituto Politécnico Nacional. Centro de Investigación en Computación. Mexico City.

Patyra, M. J.; Mlynek, D.M.; "*Fuzzy logic: implementation and applications;"* Wiley; 1996.

Oberman, S. F.; Flynn, M. J.; *"Division Algorithms and Implementations;"* IEEE Transactions on Computers; Aug 1997; Vol 46, No. 8; pp. 833–854.

Togai M.; Watanabe H.; *"Expert system on a chip: An engine for real–time approximate reasoning;"* IEEE Expert Syst. Mag., 1986, pp. 55–62, Volume 1.

Vasantha Rani, S.P.J.; Kanagasabapathy, P.; Sathish Kumar, A.; *"Digital Fuzzy Logic Controller using VHDL;"* INDICON, 2005 Annual IEEE, 11–13 December 2005, pp. 463–466.

Singh, S.; Rattan, K.S.; *"Implementation of a fuzzy logic controller on an FPGA using VHDL;"* Fuzzy Information Processing Society, 2003. NAFIPS 2003. 22nd International Conference of the North American 24–26 July 2003, pp. 110–115.

Deliparaschos, K.M.; Nenedakis, F.I.; Tzafestas, S.G.; *"A fast digital fuzzy logic controller: FPGA design and implementation;"* Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference, 19–22 September 2005, Volume 1.

Gaona, A.; Olea, D.; Melgarejo, M.; *"Sequential Fuzzy Inference System Based on Distributed Arithmetic;"* Computational Intelligence for Measurement Systems and Applications, 2003. CIMSA '03. 2003 IEEE International Symposium, 29–31 July 2003, pp. 125–129.

Manzoul, M.A.; Jayabharathi, D.; *"Fuzzy Controller on FPGA Chip;"* Fuzzy Systems, 1992., IEEE International Conference, 8–12 March 1992, pp. 1309–1316.

SOME COMPLEXITY RESULTS CONCERNING THE NON-PREEMPTIVE 'THRIFT' CYCLIC SCHEDULER

Michael Short
Embedded Systems Laboratory University of Leicester
University Road, Leicester, U.K.
mjs61@le.ac.uk

Keywords: Embedded Systems, Non-preemptive scheduling, Feasibility Analysis, Complexity.

Abstract: Non-preemptive schedulers, despite their many perceived drawbacks, remain a very popular choice for practitioners of real-time and embedded systems. Although feasibility conditions for non-preemptive scheduling models have previously been considered in the literature, to date little attention has been paid to the non-preemptive 'thrift' (or 'TTC') cyclic scheduler. This type of scheduler differs from a standard 'cyclic executive' in that it does not allow the use of inserted idle-time, and it does not require a lookup table of task executions over the major cycle of the schedule; a feasible schedule is effectively created by assigning release times ('offsets') to the tasks. To this end, this paper seeks to address the complexity of generating a feasible cyclic schedule for such a scheduler. It will be shown that when a single set of release times is assigned to the tasks, deciding feasibility of the resulting schedule is coNP-Complete (in the strong sense); and the release time assignment problem for such a scheduler is complete for $\sum_2^p$.

1 INTRODUCTION

In many real-time embedded systems, some form of scheduler is generally used instead of a full "real-time operating system" to keep the software environment as simple as possible. In general, such systems may be designed around several basic paradigms: time-triggered or event-triggered, and preemptive or co-operative (non-preemptive). This paper is concerned with single-processor time-triggered, non-preemptive schedulers.

Such architectures have been found to be a good match for a wide range of low-cost, resource-constrained applications. These architectures also demonstrate comparatively low levels of task jitter, CPU overheads, memory resource requirements and power consumption (Pont 2001; Baker & Shaw 1989; Short et al. 2008; Burns et al. 1994). Additionally, such schedulers exhibit extremely high levels of predictability and determinism. Exploratory studies also seem to indicate better transient error recovery properties in these systems than their preemptive counterparts (Short et al. 2008).

Because of these properties, non-preemptive schedulers have proved to be extremely effective at implementing systems such as process controllers, robotics, automotive applications and other types of system in which reliability is a key design goal (e.g. Gendy & Pont 2008; Short et al. 2008; Burns et al. 1994; Pont 2001).

Although non-preemptive systems are inherently free of deadlocks and other concurrency issues by the very nature of their design (Pont 2001; Baker & Shaw 1989; Short et al. 2008), the fact that tasks cannot preempt one another introduces several complexity issues related to feasibility analysis (Garey & Johnson 1979). A feasible task set is one in which all jobs generated by all tasks in the system can be said to meet their deadlines over the lifetime of the system.

Various differing models for the implementation of a 'cyclic executive' – a basic time-triggered non-preemptive design - have been proposed over the years, along with appropriate discussion and mathematical techniques for feasibility analysis. Relatively recently, the non-preemptive 'thrift' cyclic scheduler (or simply TTC scheduler) has been proposed (Pont 2001). This scheduling algorithm essentially maintains a system of congruence's to achieve its behaviour, and is presented in a simplified form in Figure 1. As can be seen, the scheduler is driven by a periodic timer signal (typically an interrupt). When tasks are released by the scheduler, they are immediately dispatched on a first-come, first-served basis (FCFS).

```
START
tick :=0; // Initialize the clock 'tick' variable
DO(FOREVER) // Enter infinite loop
   FOR i := 1 TO n DO // n is the number of tasks
      IF(((tick – ri) mod(pi)) = 0)
         Run(ti)// Immediately execute any released task
   END FOR
   Wait_Timer(); // Wait for the next timer tick to occur
   tick := tick + inc; // Increase the time index by a prefixed increment factor
END DO
EXIT
```

Figure 1: Thrift scheduling algorithm.

As can be seen, the scheduler does not support the use of 'inserted idle time', and thus does not require a lookup table of task and process executions to be created over the duration of the major cycle, which is a fundamental requirement in a 'standard' cyclic executive (Baker & Shaw 1989; Burns et al. 1994). This fundamental difference allows the scheduler to better handle tasks with larger and non-harmonic periods, requiring only storage space *O(n)* as compared to *O(nh)* where *n* is the number of tasks and *h* is the major cycle length.

In the case of the non-preemptive thrift scheduler, generating a feasible schedule surmounts to assigning specific release times to tasks such that a 'task overrun' – an overload of the processor in any given time quantum - does not occur (Pont 2001; Gendy & Pont 2008). The time quantum is normally referred to as the scheduler 'tick' interval t_{Tick} and is generally chosen to be as large as possible given the task periods, according to (1):

$$t_{Tick} = \gcd(p_1, \ldots, p_n) \tag{1}$$

With such an arrangement, w.l.o.g. the task release times can be specified as integer multiples of t_{tick} (Pont 2001; Gendy & Pont 2008). Since it has been suggested that creating the lookup table for a 'standard' cyclic executive is NP-Hard (Burns et a. 1994), this short paper seeks to explore the complexity of the offset assignment process in the 'thrift' cyclic scheduler. In Section 2 of the paper, a simple polynomial-time sufficient condition for feasibility is developed; however it is clear that this simple condition is very pessimistic. In Section 3 of the paper, the complexity of designing a TTC schedule is then considered, and it will be shown that even verifying the feasibility of a candidate solution is coNP-Complete (in the strong sense), i.e. intractable. This Section also shows that the problem of assigning specific release times to tasks is complete for the second level of the polynomial hierarchy, i.e. NP^{NP}-Complete using the notation of Garey & Johnson (1979). Section 4 concludes the paper.

2 A SIMPLE FEASIBILITY TEST

As mentioned in Section 1, the primary condition for feasibility in a non-preemptive thrift cyclic schedule is that task overruns do not occur. Given a set of tasks τ, where each $t_i \in \tau$ can be represented by a tuple:

$$t_i = (p_i, c_i, r_i) \tag{2}$$

In which p_i is the task period, c_i is the (worst case) computation time of the task and r_i is the task release time (note that a specific relative deadline can be omitted). A necessary condition for feasibility is that the computation time of each task is less than or equal to t_{tick}, and it follows that a very simple sufficient condition for feasibility of the task set is as follows:

$$\sum_{i \in \tau} c_i \leq t_{Tick} \tag{3}$$

It is quite straightforward to see that if the summed execution times of the tasks does not exceed the tick interval, then regardless of the choice of task release times (which are integer multiples of t_{tick}) then a task overrun cannot occur. It is also trivial to observe that this condition is very pessimistic, as shown by the example in Figure 2, with the tasks having periods and execution times given by {5, 2}, {15, 2 and {15,2}; resulting in a tick interval of 5 according to (1). This Figure also highlights the effect that the choice of task release times has on feasibility. In the following Section, the complexity of this release time assignment process will be investigated.

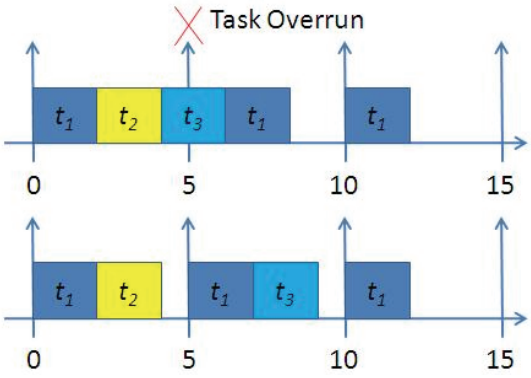


Figure 2: Effect of task release times on feasibility, showing (top): a synchronous infeasible task set; (bottom): an asynchronous feasible task set with release time of 1 tick added to t_3.

3 THE COMPLEXITY OF AN EXACT FEASIBILITY TEST

3.1 Solution Verification

Suppose we have been given a candidate solution (feasible schedule) for a thrift cyclic scheduler, i.e a set of tasks with specific release times; it is natural to consider the complexity of verifying that this solution is actually feasible. It will now be shown that deciding the feasibility of such a 'concrete' thrift schedule (the FTS problem) is strongly coNP-Complete. Membership of the problem in coNP follows since, form Figure 1, given a tick interval j in which a task overrun occurs, the execution time commanded in this interval may be determined in time proportional to n. Prior to giving the hardness proof, the simultaneous congruences problem (SCP) will be introduced; SCP was shown to be NP-Complete, in the strong sense, by Baruah et al. (1990):

Simultaneous Congruences Problem (SCP)

Instance: A set A of ordered integer pairs $\{(x_1, y_1) \ldots (x_n, y_n)\}$ and a positive integer $k > 1$.

Question: Is there a subset $A' \subseteq A$ of k ordered pairs, and a positive integer z, such that for all $(x_i, y_i) \in A'$, $z \equiv x_i \bmod(y_i)$?

Theorem 1: FTS is strongly coNP-Complete.

Proof: Transformation from the compliment of SCP. Let $\phi = < (x_1, y_1) \ldots (x_n, y_n), k >$ denote an arbitrary instance of SCP. From this a set of n concrete tasks for an instance of FTS are created according to (4), with a value of t_{tick} equal to k-1. This transformation can be performed in time proportional to n and is hence polynomial.

$$\begin{aligned} p_i &= x_i t_{tick} \\ c_i &= 1 \\ r_i &= y_i t_{tick} \end{aligned} \qquad (4)$$

Next, it is argued that a positive solution to ϕ exists iff there is a negative answer to FTS. If FTS is negative, then it implies that during at least one tick interval j, the processor demand is greater than t_{tick}. Since t_{tick} is equal to k-1, and given our construction of the task execution times, at least k tasks must be simultaneously released at the start of tick interval j; this gives a solution to ϕ with a certificate j. Conversely, if the answer to FTS is positive, then the peak processor demand is $\leq t_{tick}$ and a tick interval in which k or more tasks are simultaneously released does not exist; implying a negative answer to ϕ. Since SCP is strongly NP-Complete, the Theorem is proved. □

3.2 Release Time Assignment

Given this result, it is clear that if the verification of a candidate solution is coNP-Complete, then the problem of assigning release times (the 'TS' problem) is strongly coNP-Hard. However, since there also seems to be an exponential number of possible start times for a thrift scheduling instance, under the assumption that P ≠ NP it is also worthwhile investigating exactly where this problem lies on the so-called 'polynomial hierarchy' (Garey & Johnson 1979). It can be seen that the task release times for 'Yes' (feasible) instances of this problem can be encoded in a number of bits that is less than or equal to the task set periods, and hence the size of the overall TS problem instance. Given the previous Theorem, the resulting schedule is verifiable in polynomial time by a Turing machine with an oracle for the FTS problem; the problem resides in Σ_2^p. To show that the problem is complete for this complexity class, the Periodic Maintenance Scheduling Problem (PMSP) is now introduced. This problem is known to be Σ_2^p – Complete (Baruah et al. 1990).

PERIODIC MAINTENANCE SCHEDULING PROBLEM (PMSP)

Instance: A set C of ordered pairs $\{(n_1, c_1) \ldots (n_n, c_n)\}$, with each c_i representing a maintenance activity having an integer period n_i, positive integer $k > 1$.

Question: Is there a mapping of the activities in C to positive integer time slots such that successive occurrences of each c_i are exactly n_i time slots apart, and no more than k activities ever collide in a single slot?

Theorem 2: TS is Σ_2^p - Complete.

Proof: Transformation from PMSP.

Let $C = < ((c_1, n_1) \ldots (c_n, n_n), k >$ denote an arbitrary instance of PMSP. From this a set N of n tasks to be scheduled by TS are created according to (5), with a value of t_{tick} equal to k:

$$p_i = n_i t_{tick} \qquad c_i = 1 \tag{5}$$

Again this transformation can be performed in polynomial time. Next, it is argued that a solution to C exists iff N can be scheduled by TS. If the answer to this instance of TS is 'Yes', this implies that release times can be assigned to each task in N such that a task overrun does not occur, which from the transformation given implies that a maintenance schedule for C - in which no more than k activities occur simultaneously - also exists with a certificate $(r_1 \ldots r_n)$. Conversely, if the answer to TS is 'No', then a schedule in which a task overrun does not occur does not exist for any combination of task release times, implying that a maintenance schedule for C - in which no more than k activities occur simultaneously - does not exist. Since PMSP is Σ_2^p - Complete, the Theorem is proved. □

4 CONCLUSIONS

This paper has considered the complexity of generating a feasible cyclic schedule for the non-preemptive 'thrift' cyclic scheduler. It has been shown that an efficient - but also very pessimistic - sufficient feasibility condition exists. However, an exact solution to the problem requires the assignment of specific release times to the tasks; the complexity of this problem has been shown to be NP^{NP}-Complete, with the verification of a candidate solution being strongly coNP-Complete.

These results imply that heuristic techniques that may be used to solve the TS problem - such as the algorithm proposed by Gendy & Pont (2008) – must be strongly coNP-Hard if they provide any confirmation that the resulting schedule is indeed feasible. In conclusion, although the thrift cyclic scheduler possesses many desirable qualities, extreme computational difficulties may occur when designing a schedule. In such cases, then the non-preemptive EDF scheduling algorithm (shown to be optimal among the non-idling non-preemptive scheduling strategies by Jeffey et al. 1991) may be beneficial.

REFERENCES

Baker, T.P. and Shaw, A., 1989. The cyclic executive model and Ada, Real-Time Systems, Vol. 1, No. 1, pp. 7-25.

Baruah, S.K., Rosier, L.E. and Howell, R.R., 1990. Algorithms and Complexity concerning the preemptive scheduling of periodic tasks on one processor, Real-Time Systems, Vol. 2, No. 4, pp. 301-324.

Burns, A., Hayes, N. and Richardson, M.F., 1995. "Generating Feasible Cyclic Schedules", Control Engineering Practice, Vol. 3, No. 2, pp. 151-162.

Garey, M.R. and Johnson, D.S., 1979. Computers and Intractability: A guide to the Theory of NP-Completeness, W.H. Freeman & Co Ltd, April 1979.

Gendy, A.K. and Pont, M.J., 2008. Automatically configuring time-triggered schedulers for use with resource-constrained, single-processor embedded systems, IEEE Trans. on Industrial Informatics, Vol. 4, No. 1, pp. 37-45.

Jeffay, K., Stanat, D.F. and Martel, C.U., 1991. On non-preemptive scheduling of periodic and sporadic tasks, In Proceedings of the 12th IEEE Symposium on Real-Time Systems, pp. 129-139.

Pont, M.J., 2001. Patterns For Time Triggered Embedded Systems, ACM Press / Addison Wesley.

Short, M., Pont, M.J. and Fang, J., 2008. Exploring the impact of preemption on dependability in time-triggered embedded systems: A pilot study, In: Proceedings of the 20th Euromicro conference on real-time systems (ECRTS 2008), Prague, Czech Republic., pp. 83-91, 2-4 July 2008.

FUZZY CONTROL FOR CIRCULAR INVERTED PENDULUM

Alan Hood[1] and Umut Avci[2]
[1]School of Computing and Creative Technologies, University of Abertay Dundee, DD1 1HG, Scotland, U.K.
[2]Deptartment of Software Engineering, Izmir University of Economics, Sakarya Street No:156, Balcova, Izmir, Turkey
[1] A.Hood@abertay.ac.uk, [2]umut.avci@ieu.edu.tr

Keywords: Inverted pendulum, Fuzzy Control, Non-linear Systems.

Abstract: This paper offers alternative solutions to existing problems in inverted pendulum designs. Three main problems can be listed as limited track length, hardware complexity and inappropriate control algorithm usage. Circularly moving cart was proposed to provide unlimited track length. In order to reduce the hardware complexity, the system was controlled by a microcontroller. Finally, fuzzy control was used for the control algorithm because of its efficiency, robustness and simplicity. Efficiency of the system is presented by carrying out experiments on developed inverted pendulum control system.

1 INTRODUCTION

Inverted pendulum, used as a control system throughout the study, is composed of a rod attached to a moving cart. Aim of the inverted pendulum is to stabilize the pendulum vertically by moving the base whose action is determined by the control algorithm. One can find different designs for inverted pendulum in the literature like supported inverted pendulum (Joldis, 2006), dual inverted pendulum (Lundberg, Roberge, 2003) or 360 degree inverted pendulum (Tsai, Lin, 2003).

Main methods used for control can be listed as PID, genetic algorithms, rule-based learning, neural networks and fuzzy control (Gupta, Sinha, 1995). Fuzzy control differentiates from others because it facilitates the modeling of complex systems as in real world. In addition to modeling, design and simulation of these processes can be acquired by fuzzy logic without using complicated mathematical models because it uses linguistic variables like in real world to represent the systems.

(Mirza, Hussain, 1998) developed an inverted pendulum using PID modes for non-linear systems. (Wang et al. 2004) proposed a method to balance a rotary inverted pendulum in the minimum time. Fuzzy control was used in rotary pendulum balancing problem by (Yurkovich, Widjaja, 1996). Swing up and balancing control were applied for several phases of the process. It was seen that fuzzy logic provided better solution to control problem when system constraints were fully determined. (Ray, Das, Tyagi, 2005) used a similar approach to balance inverted pendulum moving in a limited track.

Most of these studies intend only balancing pendulum without considering all limitations like lack of memory space, excessive hardware parts and finite track length. For this reason, this study focuses on designing an ideal control system with appropriate control algorithm to overcome the abovementioned problems. The main objectives of the study are as follows:

- Building an inverted pendulum system with minimum hardware configuration.
- Determining the effects of control algorithm on system control.
- Designing inverted pendulum to provide unlimited track length.

2 SYSTEM COMPONENTS

In this section, two main parts forming the inverted pendulum system, control method and hardware, will be introduced in detail.

2.1 Control Method

Fuzzy controller comprises four main parts: rule-base, fuzzification, interface mechanism and defuzzification.

Error, difference between vertical and actual position of pendulum, has major impact on control. So, we can assign error as our first input. Change in error defines how fast pendulum changes its position, which is another important factor for control. Then, change in error can be assigned as second input.

Rules include knowledge of how system works. Each linguistic variable takes linguistic values that change from negative large to positive large. Since we have two inputs and an output it is easy to list all the rules in a tabular form as follows. One can create the tabular form for 3 and 7 linguistic values.

" PWM "		"change-in-error"				
		0	1	2	3	4
"error"	0	4	4	4	3	2
	1	4	4	3	2	1
	2	4	3	2	1	0
	3	3	2	1	0	0
	4	2	1	0	0	0

Figure 1: Rule-base with 25 rules.

In Figure 1, body represents the linguistic values for output (PWM force), left column and top row represent linguistic values for Error and Change in Error respectively. Numbers represent the magnitude of the linguistic values, i.e. "0" for "neglarge", "2" for zero and "4" for "poslarge".

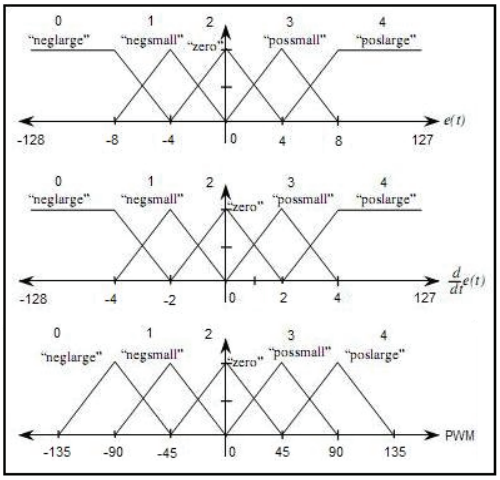


Figure 2: Fuzzy membership functions for 5 linguistic values.

Mamdani type system was selected for inference systems where minimum of the active membership degrees were operated. Center of Gravity (COG) defuzzification method was used because the universe of discourse of the output PWM(force) was continuous. As a result of this calculation, a crisp output is produced which is the force to be applied to DC motor at a time.

2.2 Hardware

In this stage, construction of the inverted pendulum control system will be explained. Input signals will be obtained from a potentiometer attached to the pendulum. According to the inputs, the microcontroller computes the control action and transfers it to the manipulating element which is a DC motor in our situation. Basic control operation can be summarized in a number of steps. The analog input from the potentiometer must be converted into a digital word in order to be processed. The microcontroller inputs these signals periodically via its internal analog-to-digital converter and calculates the required output. The electrical power supplied to motor is determined by a pulse-width modulated voltage derived using the Pulse Width Modulation (PWM) peripheral within the microcontroller. The current available from the microcontroller is insufficient to drive the motor directly so an H-bridge motor driver IC must be used. Since the motor driver system operates at a higher voltage than the microcontroller and a single power supply is to be used, a step-down voltage regulator was employed to power the microcontroller system.

Figure 3: Geared DC motor (top) and Potentiometer (bottom).

In order to provide unlimited track length, a circularly moving inverted pendulum trolley was designed in the shape of a triangle as shown in Figure 5. Control board to be attached on pendulum platform was a 9cm x 9cm stripboard comprising a 40 pin PIC 18F4620 microcontroller, LM7805

voltage regulator and L6203 H-bridge motor driver. Using all these parts in such a small space enabled us achieving minimum hardware object.

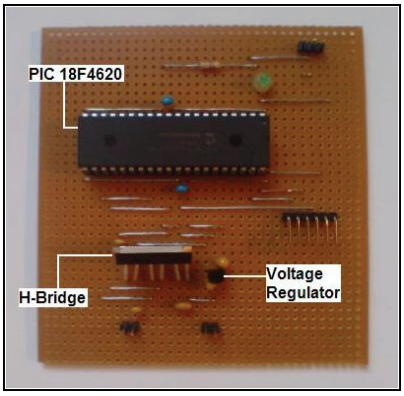


Figure 4: Control board.

The control board was used for three main operations. First, voltage was regulated in order to be used for different components throughout the control board. Second, error was read from the pot and transferred to microcontroller for processing. And lastly, PWM output produced by microcontroller was processed in H-bridge for driving the motor.

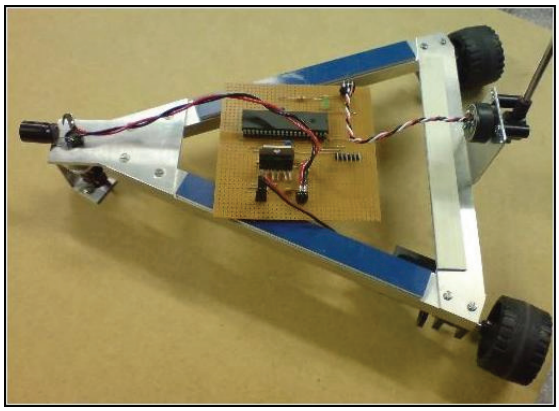


Figure 5: Side view of the pendulum trolley.

3 PERFORMANCE EVALUATION

For performance evaluation, three different fuzzy controllers defined in previous phase were embedded into inverted pendulum system one at a time. Effects of variation in control algorithm were observed experimentally in terms of system oscillation, balance angles and steady state.

Number of membership functions and rule base to be applied should be selected carefully such that system provides low PWM output for small error range and high PWM output for large error range. In the first design, three membership functions for each inputs and output were defined as well as 9 rules. For this design, system managed to lift the pendulum up by applying high pulse to motor for large errors. On the other hand, pendulum could not be balanced due to high oscillation because control action produced large PWM output even for small errors.

Fuzzy inference system with seven membership functions and 49 rules gave opposite results. Control output was small enough to balance the pendulum in very limited error angle. If pendulum starts to move away from balance limits, motor power is slowly increased by control action to keep the pendulum within the boundaries. Nonetheless, pendulum falls down to extreme edges due to inadequate acceleration.

Final fuzzy inference system design was composed of five membership functions and 25 rules. Advantages of the first two systems were included in the last version that was capable of balancing pendulum for small angles and lifting it up for relatively large angles.

	Number of Membership Functions		
	3	5	7
Lifting up angle (in degrees)	45°-50°	45°-50°	20°-25°
Oscillation in terms of angle (in degrees)	45°-50°	2°-5°	2°-10°
Steady State(S) or Dead-Band(D)	None	Yes (S or D)	Yes(S)

Figure 6: Performance comparison in terms of fuzzy membership functions.

Let us look at the situation where pendulum was balanced while platform was rotating. Main reason of this is narrow dead-band which can be defined as angle range at which pendulum is balanced and motor stops. For the project, dead-band was automatically created as narrow based on the definitions of membership functions. Since dead-band was small, pendulum was able to stay steady at the points close to the range. Thus, balance is achieved while pendulum platform rotates.

Circular movement of pendulum platform is a

good way for lifting the pendulum up only if enough acceleration is provided which is dependent on sudden change of speed. Acceleration rate can be changed by adjusting sampling time of Error and Change in Error. Longer sampling time results with higher alteration rate. Nevertheless, longer sampling time has negative effects on system control. If one selects sampling time close or bigger than falling time, system will not stabilize the pendulum. Besides, higher sampling time causes oscillation in the system between the positive and the negative sides of the vertical because of the magnitude of the PWM output. To solve this dilemma, appropriate sampling time must be selected. During inverted pendulum development process, several artificial sampling times (30 ms, 60 ms, 125 ms, 250 ms and 500 ms) were created using delay function. Test results indicated that oscillation decreases for smaller sampling times. That's why, for inverted pendulum system the fastest internal oscillator option was employed without any delay function to provide the shortest calculation interval possible.

4 CONCLUSIONS

In this paper, we focus on developing an inverted pendulum with the aim of minimizing the size and the cost of the system structure while increasing the reliability and performance. By using a microcontroller, hardware requirements are greatly reduced as well as total cost. A new approach to inverted pendulum design is proposed so that unlimited circular movement is assured. Fuzzy inference system with five triangular membership functions and 25 rules provide an appropriate way of control as far as low memory capacity of the microcontroller is concerned. Empirical results show that pendulum is balanced vertically either in steady-state or in dead-band.

Inverted pendulum control system can be improved in a number of ways. Lifting the pendulum up from extreme edges can be achieved by replacing the current pendulum with a lighter and/or shorter one. A mass can be added at the top of the pendulum in order to provide steady movement and to facilitate the stabilization but such a modification changes the place of center of gravity to a great extent. Type of the motor and size of the wheels affect the acceleration. Motor can be changed with a powerful one or/and size of the wheels can be bigger to provide more speed and acceleration correspondingly.

It is known that oscillation can be reduced by decreasing sampling time. Lower sampling time can be achieved by using higher processor speeds. In the project, sampling time was determined by the 8 MHz internal oscillator. So, a crystal oscillator may be added to system as external clock source. Such a modification increases the processor speed from 8 MHz up to 40 MHz.

REFERENCES

Gupta, M. M., Sinha N. K. 1995. *Intelligent Control Systems*. New Jersey. IEEE Press.

Joldis, A. 2006. Supported inverted pendulum, another kind of inverted pendulum. *IEEE International Conference on Automation, Quality and Testing, Robotics*. Vol. 1. pp. 143-148.

Lundberg, K. H., Roberge, J. K. 2003. Classical dual-inverted-pendulum control. *Proceedings. 42nd IEEE Conference on Decision and Control*. Vol. 5. pp. 4399 - 4404.

Mirza, A., Hussain, S. 1998. Robust controller for nonlinear and unstable systems: inverted pendulum. *Journal of AMSE*. Vol. 55. No. 3,4.

Ray, G., Das, S. K., Tyagi, B., 2005. Stabilization of Inverted Pendulum via Fuzzy Control, *IE(I) Journal–EL*.

Tsai, F. K., Lin, J.S. 2003. Backstepping Control Design of 360-Degree Inverted Pendulum Systems. *Automatic Control Conference*. pp. 138 – 143.

Wang, Z., Chen, Y., Fang, N., 2004. Minimum-time swing-up of a rotary inverted pendulum by iterative impulsive control. *Proceedings of the 2004 American Control Conference*. Vol. 2. pp. 1335 – 1340.

Yurkovich, S., Widjaja, M., 1996. Fuzzy controller synthesis for an inverted pendulum system. *Control Engineering Practice*. Vol. 4. Issue 4. pp. 455-469.

Microchip PIC18F4620 Data Sheet. [online].
Available from:
http://ww1.microchip.com/downloads/en/DeviceDoc/39626e.pdf

A MULTIOBJECTIVE CONSTRAINT OPTIMIZATION MODEL FOR MULTIMODE REPAIR PLANS

I. Barba, C. Del Valle and D. Borrego
Dpto. Lenguajes y Sistemas Informáticos, Universidad de Sevilla, Spain
irenebr@us.es, carmelo@us.es, dianabn@us.es

Keywords: Planning, Scheduling, Constraints, Repair, Multiobjective optimization, Multimode project scheduling.

Abstract: This work presents a constraint based model for the planning and scheduling of disconnection and connection tasks when repairing faulty parts. The problem involves the ordering and the selection of tasks from a set of alternatives. The goal of the plan is the minimization of the repair duration and cost. The model considers, apart from the durations, costs and resources for the tasks, the necessary delays and costs due to auxiliary tasks. The tasks can be executed in several operating modes, each one using a different machine or configuration and possibly different duration and cost. The set of all feasible plans are represented by an extended And/Or graph.

1 INTRODUCTION

There is an increasing interest for integrating planning and scheduling since real-world problems involve both of them (Boddy et al., 2004). Some of the applications involving such issues are maintenance and repair planning, where there may be a cascading set of choices which affect different features of the plan, such as duration or cost (Smith et al., 2000). Many problems can involve multiple conflicting objectives (Deb, 2008) that should be considered at the same time.

This work presents a CSP (Constraint Satisfaction Problem) model for solving a planning problem corresponding to the optimal sequencing of disconnection and connection tasks for repairing faulty components. The objective is the minimization of the total repair time and cost, considering different factors that can have an influence on it. The tasks can be executed in several operating modes (Kolisch and Drexl, 1999), each one using a different machine or configuration, and possibly different duration and cost.

The rest of the paper is organized as follows: Section 2 details the considered repair problem, Section 3 states the CSP model for planning the reparation of faulty components and, finally, Section 4 presents some conclusions and future work.

2 REPAIR PLANNING

In order to repair a faulty part, a sequence of disconnection tasks must be executed to get it, then a repair action would repair the component, and lastly some connection tasks must reconnect the system.

The use of And/Or graphs (Homem de Mello and Sanderson, 1990) allows to represent the set of all feasible plans in a natural way. An important advantage of And/Or graph is that it shows the tasks that can be executed in parallel (Fig. 1). Furthermore, both precedence constraints and those related to the selection of tasks for obtaining a correct plan, can be easily obtained from this representation.

A feasible repair plan can be seen as a set of tasks that have to be executed, containing: *Connection/Disconnection tasks*, that are executed on an established machine with a particular configuration and *Auxiliary tasks* (set-up operations, that change the configuration of a machine when two successive tasks with different configuration use it; and transportation operations, that transport the subsystems between machines when the machine where the subsystem is obtained is different from the one where is required).

The tasks can be executed in more than one operating mode, each one using a different machine or configuration and possibly different duration and cost, and will be represented by different And nodes (i.e. T_2' and T_3' in Fig. 1).

A repair graph is a subgraph of the And/Or graph which only contains the tasks that could be necessary to repair some components. Another important consideration is that a connection (disconnection) task T is reversible if its corresponding disconnection (connection) task T' is feasible. The model supposes two

assumptions: (A1) All tasks are reversible and (A2) Subsystems that do not include the faulty components are not disconnected. Taking into account this, other subsystems different from the ones generated by the disconnection process can appear in the connection process. Moreover, disconnection tasks only handle subsystems that contain the faulty component. Although the disconnection process is linear, the connection can contain tasks that may execute in parallel with others.

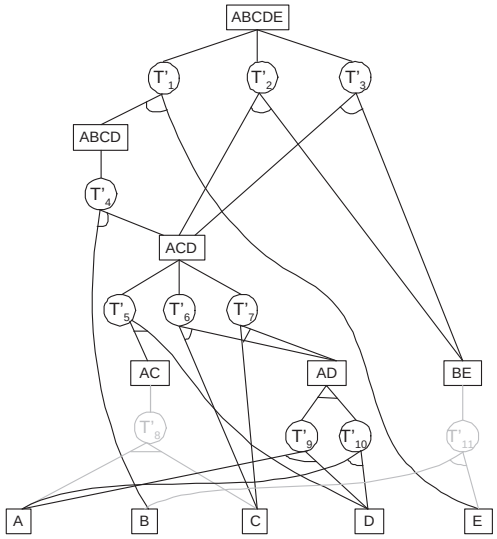


Figure 1: A simplified disconnection And/Or graph.

3 THE CSP MODEL

Constraint Programming (CP) has been evolved in the last decade to a mature field because, among others, of the use of different generic and interchangeable procedures for inference and search, which can be used for solving different types of problems (Rossi et al., 2006). Once a problem is modelled by a CSP, a generic or specialized CSP solver can be used in order to obtain the required solution.

Taking into account (A1) and (A2), the graph can be simplified removing those And nodes below the Or nodes corresponding to subsystems which do not contain the faulty part (Fig. 2).

3.1 Variables of the CSP

Selection Variables. For each And node, two boolean variables represent if the connection $s(T)$ and disconnection tasks $s(T')$ are selected for the solution. Furthermore, for each Or node, two boolean variables represent if the subsystem S appears in the connection $s(S)$ and disconnection processes $s'(S)$.

Resource Variables. For each And node, $M(T)$ and $M(T')$ represent the machines used, and $Cf(T)$ and $Cf(T')$ are the necessary configuration on them for the connection and disconnection tasks respectively. On the other hand, the machine where a subsystem is obtained after the corresponding disconnection and connection task, are represented by the variables $m'(S)$ and $m(S)$.

Time Variables. For each And node, the durations of the associated tasks $Dur(T)$ and $Dur(T')$ are established. For the auxiliary operations, $\Delta_{cht}(M,Cf,Cf')$ denotes the time needed for changing the configuration of the machine M from Cf to Cf', and $\Delta_{mov}(S,M,M')$ denotes the time needed for transporting the subsystem S from machine M to machine M'. Finally, a temporal delay $\Delta_{subst}(C)$ is associated to the reparation of the faulty component. On the other hand, for each And node: its starting and ending times, $t_i(T)$ and $t_i(T')$, $t_f(T)$ and $t_f(T')$. For each Or node, the times when it is obtained after connection, $t_{OR}(S)$, and disconnection, $t'_{OR}(S)$.

Cost Variables. For each And node: its connection $Cost(T_i)$ and disconnection cost $Cost(T'_i)$. Regarding to the auxiliary operations, $Cost_{cht}(M,Cf,Cf')$ denotes the cost of changing the configuration of the machine M from Cf to Cf', and $Cost_{mov}(S,M,M')$ denotes the cost of transporting the subsystem S from machine M to machine M'. Furthermore, a cost $Cost_{subst}(C)$ is associated to the reparation of the faulty component. On the other hand, for each And node, the selection of the corresponding task T may be associated some additional costs: first, the variable $cost_{mov}(T_i)$ represents the possible costs associated to the movement of subsystems; and secondly, the variable $cost_{cht}(T_i)$ represents the possible costs of change of configuration.

The original And/Or graph has been extended, adding new types of links between And nodes. The new links represent non-precedence constraints: due to the use of shared resources by the tasks and due to the change of configurations in the machines.

Fig. 2 shows the extended and simplified repair And/Or graph of a system consisting in *ABCDE* components when substituting *D*. A typical objective for such a problem would be the minimization of the elapsed time of the plan, given by the variable ($t_{OR}(ABCDE)$). Another important issue is the total cost of the complete repair plan. In this work, a multi-objective optimization is pursued, encompassing both of them.

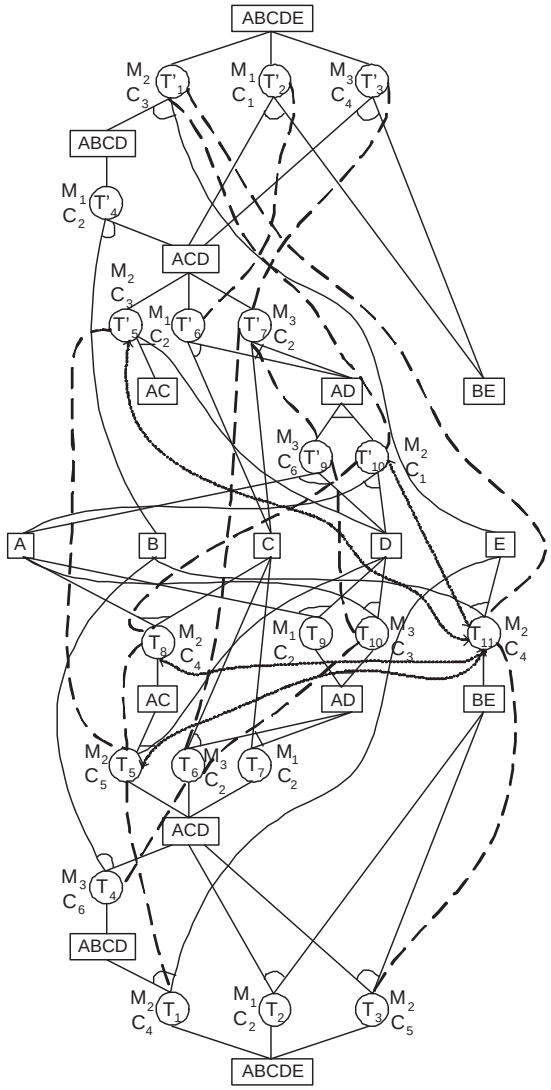


Figure 2: The simplified repair And/Or graph with relations (5) and (6) between tasks

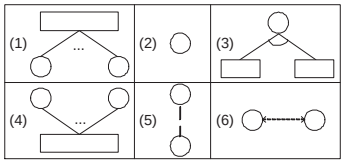


Figure 3: Types of Relations

3.2 Types of Constraints

Six types of relations are considered (Fig. 3), each one representing a link or component of the extended graph (Fig. 2): *Relations of type (1)* collect the relation between the information from an Or node and the And nodes below it. *Relations of type (2)* consider the durations of tasks, and correspond to the relations between its starting and ending times. *Relations of type (3)* collect the relation between the information from an And node and the Or nodes below it. *Relations of type (4)* consider the relation between the selection of an Or node and all the And nodes above it. *Relations of type (5)* are due to the delay needed for a change of configuration in a machine between the executions of two successive tasks using the same machine with precedence constraints among them. *Relations of type (6)* consider the relation between some tasks that use the same resource. Types (1), (2), (3) and (4) come from the relations between the nodes included in the original And/Or graph, while types (5) and (6) come from the use of (same or different) resources by the different tasks, and they are related to new links between tasks in the extended And/Or graph.

Taking into account the variables of the proposed model, selection, resource, time and cost constraints are considered. A previous work (Del Valle et al., 2009) includes the first three kinds of constraints.

Selection Constraints. Collect the relations between the boolean variables that represent if the tasks are selected for the solution and the subsystems appears in the repair process. A special case is for the complete system and for the faulty component, which always will be part of the solution. The constraints of type (1) include the selection of disconnection tasks T' and connection tasks T with that of subsystems: $s'(S) \Leftrightarrow XOR_{T'_i \in succ(S)}(s(T'_i))$ and $s(S) \Leftrightarrow XOR_{T_i \in succ(S)}(s(T_i))$. Related to relations of type (3), the selection of the two Or nodes if the And node is selected: $s(T') \Rightarrow s'(S_1) \wedge s'(S_2)$ and $s(T) \Rightarrow s(S_1) \wedge s(S_2)$. The constraints of type (4) include the selection of tasks T' and T with that of subsystems: $s'(S) \Leftrightarrow XOR_{T'_i \in pred(S)}(s(T'_i))$ and $s(S) \Leftrightarrow XOR_{T_i \in pred(S)}(s(T_i))$.

Resource Constraints. Consider the relations between the machines used in the tasks, and those where the subsystems are obtained after them. Related to relations (1), the machine m where a subsystem is generated after a connection task is the machine used by this task: $s(T_i) \Rightarrow m(S) = M(T_i)$. Related to relations of type (3), the machine m' where a subsystem is generated after a disconnection task is the machine used by this task: $s(T'_i) \Rightarrow m'(S_1) = m'(S_2) = M(T'_i)$.

Time Constraints. Time constraints collect the relations between the start and the end times of the tasks, and the time when the subsystems are obtained. Related to relations of type (1), these constraints establish the disconnection t'_{OR} and connection times t_{OR} of Or nodes related to the start times of the discon-

nection tasks or the end times of the connection tasks: $s(T_i') \Rightarrow t_i(T_i') \geq t'_{OR}(S) + \Delta_{mov}(S, m'(S), M(T_i'))$ and $s(T_i) \Rightarrow t_f(T_i) = t_{OR}(S)$. Related to relations of type (2), these constraints consider the end time of the tasks related to the start time and the durations of them: $s(T_i') \Rightarrow t_f(T_i') = t_i(T_i') + Dur(T_i')$ and $s(T_i) \Rightarrow t_f(T_i) = t_i(T_i) + Dur(T_i)$. Related to relations (3), the next constraints include the equality constraint between the disconnection times of the Or nodes t'_{OR} and the end time of a disconnection task T' above them: $s(T_i') \Rightarrow t_f(T_i') = t'_{OR}(S_1) = t'_{OR}(S_2)$, and the precedence between the connection time of the Or nodes t_{OR} and the start times of connection task T, and considering the possible delays due to the transportation of subsystems if the two successive tasks involving it use different machines: $s(T_i) \Rightarrow t_i(T_i) \geq t_{OR}(S_1) + \Delta_{mov}(S_1, m(S_1), M(T_i))$ and $s(T_i) \Rightarrow t_i(T_i) \geq t_{OR}(S_2) + \Delta_{mov}(S_2, m(S_2), M(T_i))$. Related to relations of type (5), these constraints establish that for a task T_i, and its closest predecessor task T_j using the same machine m, taking into account the possible change of configuration: $(s(T_i) \wedge s(T_j)) \Rightarrow t_i(T_j) \geq t_f(T_i) + \Delta_{cht}(m, Cf(T_i), Cf(T_j))$. For each two tasks T_i and T_j requiring the same machine m, with no precedence constraint among them, and which may belong to the same repair plan, the constraints of type (6) express the two possible orders of execution of the tasks: $(s(T_i) \wedge s(T_j)) \Rightarrow (t_i(T_i) \geq t_f(T_j) + \Delta_{cht}(m, Cf(T_j), Cf(T_i)) \vee \ t_i(T_j) \geq t_f(T_i) + \Delta_{cht}(m, Cf(T_i), Cf(T_j)))$. For the *Or* leaf nodes t'_{OR} and t_{OR} are equals, except for the faulty component due to the delay corresponding to the reparation.

Cost Constraints. The cost of a plan can be established by the aggregated costs associated to the execution of the selected tasks. The total cost of selecting a task T_i involves:

- the execution cost of the task, $Cost(T_i)$
- the cost associated to the possible machine movement of one or two subsystems, $cost_{mov}(T_i)$: first, in disconnection tasks T_i', the possible movement of the subsystem related to the Or nodes above it, related to relation (1), $cost_{mov}(T_i') = Cost_{mov}(S, m'(S), M(T_i'))$; secondly, in connection tasks T_i, the possible movement of the two subsystems related to Or nodes below it, related to relation (3), $cost_{mov}(T_i) = Cost_{mov}(S_1, m(S_1), M(T_i)) + Cost_{mov}(S_2, m(S_2), M(T_i))$.
- the possible cost associated to a change of configuration on $M(T_i)$, $cost_{cht}(T_i)$. If $M(T_i)$ has been used before by another task with a different configuration, it is necessary to change it. The cost of the change of configuration depends of the sequence of tasks for each machine, so there must be considered the **precedent task** executed on $m(T_i)$.

Taking into account this, $cost_{cht}(T_i) = Cost_{cht}(M(T_i), Cf(PM(T_i)), Cf(T_i))$, where $PM(T_i)$ is the precedent task executed on $m(T_i)$. Also, the total cost of a plan can be defined as $\sum_{T_i} s(T_i)(Cost(T_i) + cost_{mov}(T_i) + cost_{cht}(T_i))$.

4 CONCLUSIONS AND FUTURE WORK

This work proposes a CSP model for the planning and optimal sequencing of disconnection and connection multi-mode tasks when repairing faulty components, taking into account the minimization of time and cost. The proposed model can be solved using conventional methods for a generic CSP. As future work, it is intended to use different strategies to solve the problem, working with heuristic algorithms based on the resulting state of the constraint propagation process and on the objective functions to be optimized.

ACKNOWLEDGEMENTS

This work has been partially supported by the Spanish Ministerio de Educación y Ciencia through a coordinated research project (Grant DIP2006-15476-C02-01) and Feder (ERDF).

REFERENCES

Boddy, M., Cesta, A., and Smith, S. (2004). *ICAPS-04 Ws. Integrating Planning into Scheduling.* AAAI Press.

Deb, K. (2008). Introduction to evolutionary multiobjective optimization. *Lecture Notes in Computer Science 5252 LNCS*, pages 59–96.

Del Valle, C., Márquez, A., and Barba, I. (2009). A csp model for simple non-reversible and parallel repair plans. *Journal of Intelligent Manufacturing. To appear (DOI 10.1007/s10845-008-0162-9)*.

Homem de Mello, L. and Sanderson, A. (1990). And/or graph representation of assembly plans. *IEEE Transactions on Robotics and Automation*, 6(2):188–189.

Kolisch, R. and Drexl, A. (1999). Local for multi-mode resource-constrained project. *IIE Transactions (Institute of Industrial Engineers)*, 29(11):987–999.

Rossi, F., Van Beek, P., and Walsh, T. (2006). *Handbook of Constraint Programming*. Elsevier.

Smith, D., Frank, J., and Jónsson, A. (2000). Bridging the gap between planning and scheduling. *Knowledge Engineering Review*, 15(1):47–83.

FUZZY TRAJECTORY TRACKING FOR AN AUTONOMOUS MOBILE ROBOT

Carlos Fernández Caramés, Vidal Moreno Rodilla
Departamento de Informática y Automática, University of Salamanca, Plaza de los Caídos S/N, Salamanca, Spain
carlosfc@usal.es, vmoreno@usal.es

Belén Curto Diego, José Andrés Vicente Lober
Departamento de Informática y Automática, University of Salamanca, Plaza de los Caídos S/N, Salamanca, Spain
bcurto@usal.es, andres@usal.es

Keywords: Mobile robot, Heading sensor, Fuzzy controller.

Abstract: This paper proposes a fuzzy controller embedded in a closed-loop control system designed to make a robot track a straight line. The system uses a heading sensor to measure the error in the orientation of the robot. A real robot is simulated in Matlab so as to test and accelerate the development process of the fuzzy controller. Finally, experimental results of the simulated and the real robot are presented, showing the effectiveness of our approach under strong disturbances such as unexpected robot rotations.

1 INTRODUCTION

There is no doubt whatsoever that moving from one place to another is a must for every mobile robot. The type of movements that a robot will perform will nonetheless be different depending on if it is familiarized with its surroundings or not. When a robot is exploring an unknown environment, it will typically wander aimlessly either trying to build a map, trying to locate itself, or both things at the same time. However, when a robot is within a previously known environment, its movements will generally be planned by a high level path planner, provided that a map is available.

Path planning, together with map building and localization, is one of the three fundamental tasks a robot has to master to fully solve the navigation problem, and it is the area of navigation which has received the most attention (Murphy, 2000). The path planning problem consists in designing a path between an initial position and a target position such that (a) the robot does not collide with any static or dynamic obstacles in the environment and (b) the planned motion is consistent with the kinematic constraints of the vehicle (Zou et al., 2006). The kinematics of a vehicle are determined by the steering mechanism, being differential drive and Ackermann drive two of the most frequently used steering mechanisms for mobile robots.

There are many different approaches to path planning, both for differential and for Ackermann steered robots, but in the end, the final result of any path planner is a sequence of path segments (Baltes and Hildreth, 2001). Many planners use a sequence where each segment is either a straight line, a full left turn or a full right turn, based on the early work of Reeds and Shepp (Reeds and Sheep, 1990), which proves that the shortest path for any vehicle can be planned using exclusively these three types of segments.

Once the path is planned, the robot should be able to follow the planned segments as accurately as possible. The aforementioned maneuvers —straight lines and full turns— may seem easy to perform by a human driver with some experience, but they are not straightforward at all for an autonomous mobile robot. Tracking a straight line is somewhat difficult than tracking full turns, and this is particularly true for a differential drive robot. Moving the wheels of a differential robot at the very same speed is not enough to achieve a straight line, because different wheel radii or wheel slippage, among other reasons (see (Borenstein et al., 1996)), will cause the robot to get out of its intended trajectory sooner or later. Ackermann steered robots, although more robust for straight line tracking than differential steered robots, are also difficult to be driven along a straight line.

For example, in our robotics research group we own an unmanned forklift truck (reverse Ackermann steering), whose main difficulty is that it has a high degree of looseness in its steering wheel.

The conventional approach for controlling these robots and make them follow a straight line would be to design a PID (Proportional, Integral and Derivative) controller. Consequently, we would need to know every physical detail about the robot and its environment so as to be able to model such a system mathematically (Marzi, 2006). However, we humans do not need accurate information from the environment or from a vehicle to control it and perform successful maneuvers. Most of the time we deal with approximate reasoning rather than precise, and it can be expressed linguistically by a series of if-then rules. Luckily, this rationale is not only available humans, since fuzzy logic provides us with a mathematical framework to translate our linguistic expert knowledge into numerical data which can be used by robots.

In this paper we present a fuzzy approach to solve the problem of straight line tracking for differential robots using an Attitude and Heading Reference System (AHRS).

2 GENERAL SYSTEM DESCRIPTION

The general block diagram of our system is depicted in Fig. 1, which shows a fuzzy controller embedded in a closed-loop control system. The system works by selecting a reference heading (RH), which the robot will have to follow. Then, the MTi sensor measures the heading (MH) of our robot (AmigoBot). The difference between this two data is used to calculate the heading error (e) and its derivative ($\dot{e}$). Next, these data are used as the inputs of the fuzzy controller, which determines the change in speed (SC) needed to correct the heading (H) of the robot.

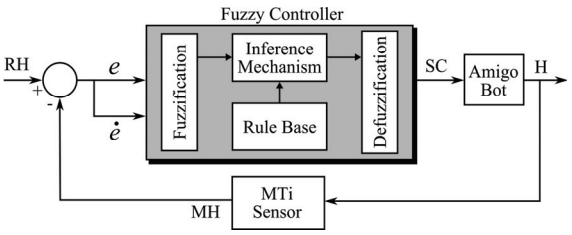


Figure 1: General system description.

The core of the system is the fuzzy controller, which consists of four components: (1) the "rule base" is the set of rules that control the system. (2) The inference mechanism evaluates which control rules are relevant. (3) The fuzzification interface modifies the inputs to the fuzzy controller so that they can be interpreted and compared to the rules in the rule base. (4) The defuzzification interface transforms the conclusions reached by the inference mechanism into the input to the robot.

The MTi is a low-cost Attitude and Heading Reference System (AHRS) from Xsens technologies. We use it to measure the heading angle of the robot. The mobile robot we have chosen is the AmigoBot from MobileRobots Inc.

3 KINEMATIC MODEL OF THE ROBOT

The kinematic model of the AmigoBot is depicted in Fig. 2. There, *ICC* stands for Instantaneous Center of Curvature; $v_l(t)$ and $v_r(t)$ are the linear velocity of the left and right wheel; R is the curvature radius described by the middle point of the wheel axis, and b is the distance between wheels. When the linear velocities of the left and right wheel are different, the robot turns around the ICC with angular velocity $w(t)$.

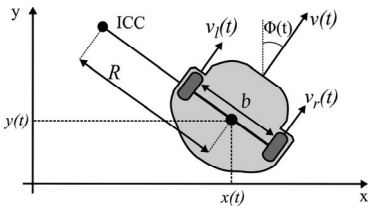


Figure 2: Kinematic model of the Amigobot.

Additionally, if we designate R_l and R_r as the curvature radii described by the left and the right wheel, respectively, then $R = (R_l + R_r)/2$. Taking this into account, and the fact that $v_l(t) = w(t) \cdot R_l$ and $v_r(t) = w(t) \cdot R_r$, the linear velocity of the robot can be expressed as $v(t) = w(t)R = \left(v_r(t) + v_l(t)\right)/2$. If we continue developing, we can obtain the angular velocity of the robot as $w(t) = \left(v_r(t) - v_l(t)\right)/b$.

The state of the robot is represented by the variables $x(t)$, $y(t)$ and $\Phi(t)$, and it can be obtained by integrating (1). We use it to study the performance of the robot in a simulated environment.

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\phi}(t) \end{bmatrix} = \begin{bmatrix} \sin(\phi(t)) & 0 \\ \cos(\phi(t)) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v(t) \\ w(t) \end{bmatrix} \quad (1)$$

4 FUZZY LOGIC CONTROLLER

The definition of a fuzzy system can be broken down in several parts (Passino and Yurkovich, 1998): a) variables and values, b) rule set and c) membership functions.

We have used three linguistic variables: "heading error", "change in heading error" and "speed correction", and 5 different linguistic values for each variable: NL (negative large), NS (negative small), Z (zero), PS (positive small) and PL (positive large). The way these terms are used is indicated in Table 1, where CW and CCW stand for clockwise and counterclockwise, respectively.

Table 1: Meaning of the linguistic terms.

	Positive	Negative
Heading error	Robot is rotated CCW	Robot is rotated CW
Change in heading error	Robot is rotating CCW	Robot is rotating CW
Speed Correction	Robot needs to rotate CCW	Robot needs to rotate CW

Table 2: Rule table for the Amigobot.

Speed Correction		Change in heading error ($\dot{e}$)				
		NL	NS	Z	PS	PL
Heading error (e)	NL	PL	PL	PL	PS	Z
	NS	PL	PL	PS	Z	NS
	Z	PL	PS	Z	NS	NL
	PS	PS	Z	NS	NL	NL
	PL	Z	NS	NL	NL	NL

Using the linguistic quantification stated in the previous subsection, we have designed a set of rules (see Table 2) that describe how to make the robot follow a straight line. Finally, the membership functions of the system are depicted in Fig. 3.

Because of a hardware limitation, the linear speed of the Amigobot wheels can only be set to multiples of 20 mm/s. Thus, we have designed the membership function of the speed correction variable taking into account this peculiarity, and hence each unit in the *x* axis means 20 mm/s. Once the fuzzy system computes an output speed correction, this value is added to the current right wheel speed and subtracted from the current left wheel speed. Thus, the required torque is achieved and the robot rotates toward the desired direction while the linear speed remains constant.

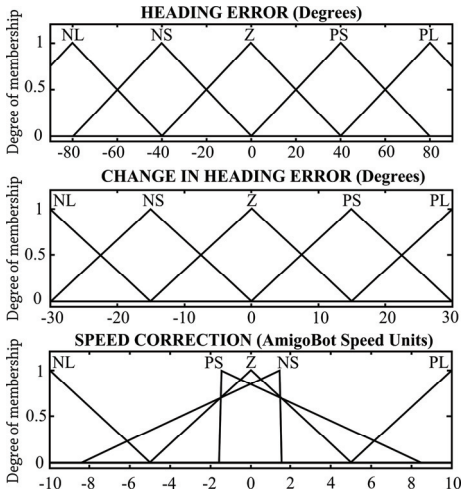


Figure 3: Membership functions.

5 EXPERIMENTAL RESULTS

We have simulated the movements of AmigoBot using the kinematic model (1) and the readings from the heading sensor, where we have added random white noise to mimic the specifications of the MTi unit. The simulation results were very similar to the real world results, and therefore, we only show the performance of the real robot. During the tests, the robot was commanded to travel at a constant speed of 0.2 m/s, and it was subjected to four strong disturbances of ≈-90º, ≈90º, ≈-45º and ≈45º.

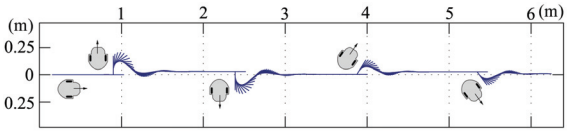


Figure 4: Simulated robot trajectory.

An illustration of the trajectory followed by the robot is shown in Fig. 4, where the robot is represented as a short segment. Next to each deliberated turn, the robot is depicted. The corresponding input and output variables for the simulation of the fuzzy system are shown in Fig. 5. As it can be seen, the robot gets stabilized quickly and smoothly after the unexpected rotations. When the robot is subjected to 45º turns, it offers fast response times (it is stabilized in 1s) and excellent performance: it does not virtually oscillate at all. On the other hand, when it is presented with strong disturbances (90º turns approximately), its response

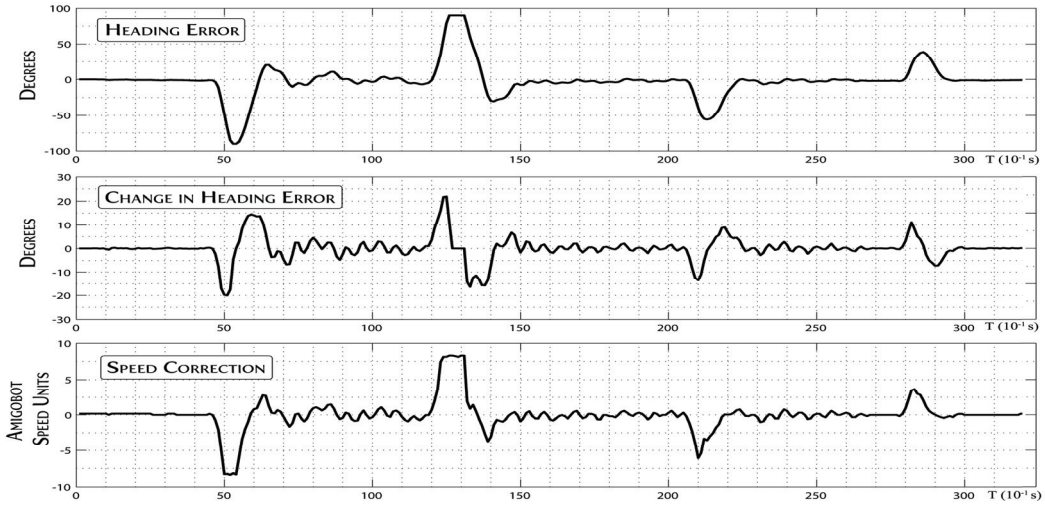


Figure 5: Real robot experimental results.

is still acceptable although the heading takes more time (2-3 s) to get stabilized because of oscillation.

6 CONCLUSIONS

Path planners are widely used when a mobile robot is within a known environment. Many path planning methods give as a result a sequence of straight line segments and full turns, because it has been proved that the shortest path between two points can be achieved that way. Taking into account that full turns are less difficult to perform, we have proposed and implemented a fuzzy controller that is capable of following straight lines under strong disturbances. The fuzzy controller is embedded in a closed-loop control system, and relies on a AHRS unit —used as a heading sensor— to guarantee that the robot faces the right direction.

The performance obtained with the real robot is quite similar to the simulation results, and the robot is capable of tracking a straight line even under unexpected turns of 90º. Although the real robot includes a nonlinearity by which the linear speed of each wheel can only be set in 20 mm/s increments, the fuzzy controller performs equally well under this circumstance.

Future work includes modifying this system and adapting it to a real forklift truck. Such a system is being tested, and it is giving excellent results in the simulation stage.

ACKNOWLEDGEMENTS

This work has been partially funded by the Spanish JCYL in conjunction with the European Social Fund, under the FPI grants program: ORDEN EDU/1453/2005 de 28 de octubre. It also has been funded by the Spanish JCYL under project ref. SA030A07, and the Spanish MEC under project ref. DPI2007-62267.

REFERENCES

Baltes, J. and Hildreth, N. (2001). Adaptive path planner for highly dynamic environments. *Lecture Notes in Computer Science*, 2019:76 – 85.

Borenstein, J., Everett, H. R., and Feng, L. (1996). Where am I? sensors and methods for mobile robot positioning. Technical report, The University of Michigan.

Marzi, H. (2006). Fuzzy control of an inverted pendulum using ac induction motor actuator. In *Proceedings of the 2006 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications*, pages 109–114.

Murphy, R. (2000). *Introduction to AI Robotics*. MIT Press.

Passino, K. M. and Yurkovich, S. (1998). *Fuzzy Control*. Addison-Wesley.

Reeds, J. and Shepp, L. (1990). Optimal paths for a car that goes both forward and backward. *Pacific Journal of Mathematics*, 145(2):367 – 393.

Zou, A.-M., Hou, Z.-G., Tan, M., and Liu, D. (2006). Path planning for mobile robots using straight lines. In *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control*, pages 204 – 208.

A NEW NEURAL SYSTEM FOR LOAD FORECAST IN ELECTRICAL POWER SYSTEMS

A Topological Level Integration of Two Horizon Model Forecasting

Rodrigo Marques de Figueiredo, José Vicente Canto dos Santos and Adelmo Luis Cechin
PIPCA - UNISINOS, Av. Unisinos 950, São Leopoldo, Rio Grande do Sul, Brazil
rdgmarques@yahoo.com.br, jvcanto@unisinos.br, acechin@unisinos.br

Keywords: Artificial Neural Networks, Electric Power Systems, Load Forecasting.

Abstract: This work presents a new integrated neural model approach for two horizons of load forecasting. First of all is presented a justification about the design of a computational neural forecasting model, explaining the importance of the load forecast for the electrical power systems. Here is presented the design of the two neural models, one for short and other for long term forecasting. Also is showed how these models are integrated in the topological level. A neural model that could integrate two forecasting horizons is very useful for electrical system enterprises. The computational system, here presented, was tested in three different scenarios, where each scenario has specific electrical load behaviour. At last the results is commented and explained.

1 INTRODUCTION

Actually the load forecasting is an important tool for energy enterprises. The forecast for electrical power systems is subject to internal variables in addition of external variables, stochastics variables, like meteorological and macroeconomic variables. The first one has an imply in residential loads and the second one has a strong imply in industrial loads (Ardil et al, 2007). The modern way to develop a forecaster is by the using of ANN, Artificial Neural Network, models.

In the literature, there are many papers about the use of neural modeling for only one forecasting horizon, examples are the work of Botha (Botha, 1998), Drezga (Drezga, 1999), Saad (Saad, 1999), Charytoniuk (Charytoniuk, 2000), Fukuyama (Fukuyama, 2002), Funabashi (Funabashi, 2002) and Abdel-Aal (Abdel-Aal, 2004). But neural modeling for two or more forecasting horizons is scare, one of the few exmples is the work of Shirvany (Shirvany, 2007).

The present paper propouses a new neural model for load forecasting by the using of two integrated models, one for short term and other for long term load forecasting. The resulting model has the ability for short and long term load forecasting at the same time, with better performance, both in response quality and computational performance.

The electrical power system focused in this forecast system is located in a large area in the south of Brazil. All the tests and results showed in this paper are referred to this system. This area is divided in seven nodes and each node has one type of the three electrical consuption behaviour, residential, industrial or a mixed type. After this introduction, follows the description of the proposed system, the tests performed and the results obtained and, finaly, our conclusions.

2 THE COMPUTATIONAL FORECASTING SYSTEM

The forecasting system consists in two neural models, one for short term and other for long term forecasting. These neural models are given by the artificial neural network application. The models were individually designed and validated to later be integrated. The data base of variables available to be used to design the models are given by meteorological, macroeconomic and electrical variables.

The variable space for an electrical system is too large, even when it is reduced for the three types showed above. For a better model response this

space must be reduced. Variable selection methods are the best way to reduce the variable space removing from model most of redundant and irrelevant variables.

2.1 Variable Selection

The variables were selected by the using of forward selection. In this method the neural model is constructed by its interaction, where in each interaction one variable is included in model. The criteria used to the model construction are the minor response error for a validation (Seeger, 2003). This algorithm runs until a stop criteria, in this paper case an error level minor than fifteen percent. For the two models, short and long term, this method is applied by individually manner. In the variable selection in addition to the inclusion of new variables were also varied the number of neurons in the hidden layer of ANNs, seeking for the best system performance.

2.2 Long Term Model

The main objective of this model is to provide the behaviour information of the electrical system to short term model, through the topological integration. In this model the forecasting horizon chosen was the monthly horizon, because that information is very important for the business of the electrical energy sector utilities (Quintanilha et al, 2005).

After the forward selection application the variables were selected, resulting in the neural model for long term forecasting. The monthly information of temperature and residential, commercial and industrial electrical load as input, with six neurons in hidden layer and one as output, indicating the long term total load forecast. This model uses as input the monthly information of one year and one day ago. That information give to the long term model the monthly tendency of each month of the year with all seasonal influences. This fact makes the model more robust.

2.3 Short Term Model

This model try, as main objective, mimetizes the electrical power system load behaviour. As like long term model, this model uses the forward selection to choose its variables. In this model case faster variables behaviour is relevant to it, like meteorological and electrical variables.

After the use of forward selection the neural model was constructed with the variables selected. This uses the daily information, about one day ago, of temperature, humidity and total electrical load as input, with six neurons in the hidden layer and one as output, representing the total load for the shot term forecast.

2.4 Model Integration

The integration of the short and long term forecast models is the main step of the computational system design. Is important keep in mind that this integration is given in the topological level. With this type of integration only the tendencies of each model are passed to the other. In other types of integration the error also is integrated.

The neurons sharing guarantee the tendencies exchange between long and short term models without polluting yours responses. But this is not a total share, only a parcel of these neurons is shared.

Using the neural models for short and long term forecasting with six neuron in hidden layer, a new neural model are created with merging these models. There were made tests to verify the number of shared neurons in hidden layer is needed to bettering the model response. In this test the number of shared neurons was varied in one to all (twelve).

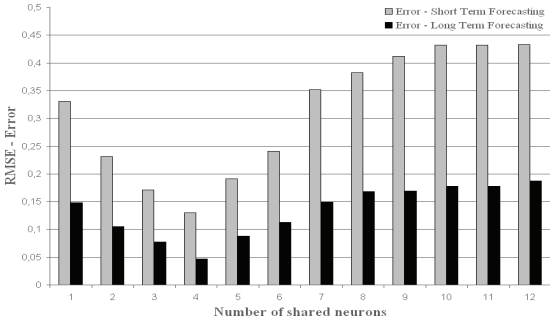


Figure 1: Trial with neurons sharing.

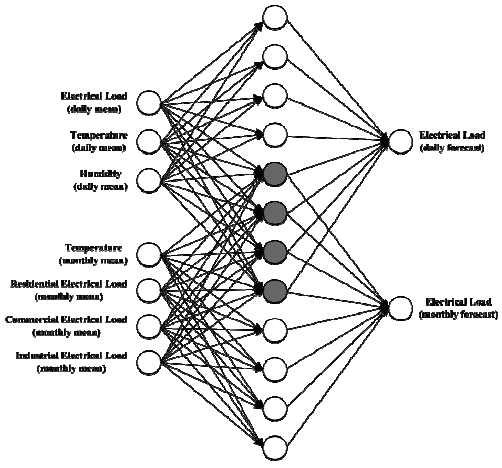


Figure 2: Neural model integration.

The Figure 1 shows that four is the best number of shared neurons to this application. The Figure 2 shows the final arrange of neural model, in highlighting the shared neurons in dark color. Also are showed the inputs and the outputs of final model.

The final model uses twelve neurons in hidden layer, with four exclusively used by short term model, four for the long term model and four neurons being shared by the two models, unifying these models in only one.

2.5 System Architecture

The architecture of the computational system is given by three main parts, or modules. This architecture is showed in Figure 3.

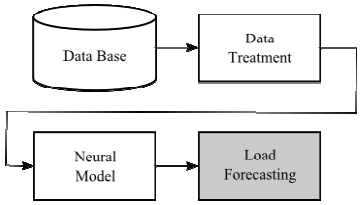


Figure 3: Computational system architecture.

Database contains all information about the electrical power system. For forecasting models is very important a large database as possible (Swinder et al, 2007). In the data treatment module the data is synchronized, normalized and separated per type. This learning occurs throughout the artificial neural network (ANN) training. The data set is delivered to the neural model aligned like is showed in Figure 4.

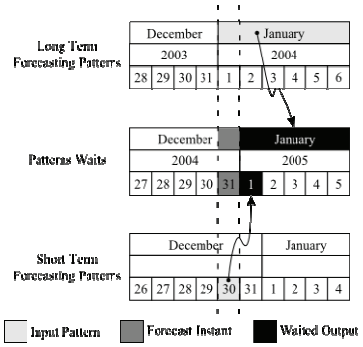


Figure 4: Data set temporal aligning.

In Figure 4 the forecast instant represents the moment when the computational forecasting system is executed. This data alignment avoids the need for not available data. That case occur when two forecast-ting horizons are used in the same model and one horizon is overridden by the other.

3 TESTS AND RESULTS

The system proposed was subjected to three different scenarios of load consumption being that, Industrial Load Region, Residential Load Region and Mixed Load Region. The tests outcomes of the integrated system are compared with the outcomes of the separated models for short and long term forecasting. In the tests was used the same number of sample for each region data set, and the same data set to individually forecaster (short and long term) and the integrated proposed system. There are performed the Ten-Fold Cross Validation method to prove the benefit of the models integration. As quantitative metric was used was the Root Mean Squared Error (RMSE), and all the results presented in this section were obtained with this metric.

3.1 Industrial Load Region Test

Industrial load has a seasonal behaviour with strong dependence of macroeconomic factors, that indicates the production behaviour of the industry and per consequence it is your electrical power consumption. The proposed system and individually models, developed to create the proposed system, results for this scenario are showed in Table 1.

Table 1: Industrial region test results.

Forecast Horizon	Propose Integrated System	Individually Models
Long Term	4,6%	21,4%
Short Term	13,2%	23,7%

3.2 Residential Load Region Test

The residential load presents a different behaviour, it is not seasonal. This type of consumer has a behaviour closely liked to the meteorological conditions. In cold days the residential consumer uses their heaters, and in the hot days they use their air conditioners. The system outcome to this type of load consumption is given in Table 2.

Table 2: Industrial region test results.

Forecast Horizon	Propose Integrated System	Individually Models
Long Term	6,7%	22,9%
Short Term	13,0%	24,8%

3.3 Mixed Load Region Test

Mixed load consumer regions are areas where there

is a balance between residential and industrial consumers. In those areas there is no definition about the load behaviour, because it follows the trend given by the industrial and residential load. The system outcome to the mixed type of load consumption is given in Table 3.

Table 3: Industrial region test results.

Forecast Horizon	Propose Integrated System	Individually Models
Long Term	5,5%	22,1%
Short Term	11,7%	24,6%

In Figure 5 is ploted the results for short term forecast, comparing the pattern wait with outcomes of conventional forecasting system and the new neural system proposed in this paper. Note that the proposed system (represented by solid black line) fits perfectly with the pattern waited (grey line), the conventional neural system, represented by the short term model (dashed line) before developed has a worst behaviour.

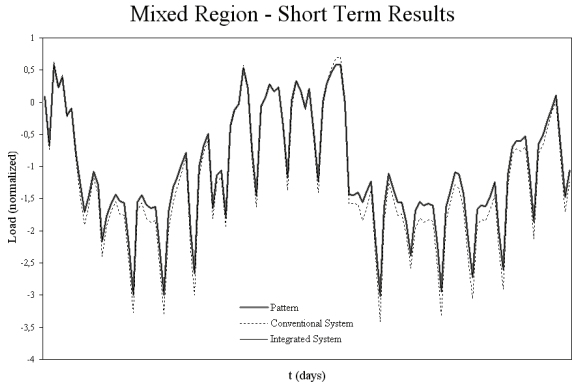


Figure 5: Short term load forecasting for mixed region.

4 CONCLUSIONS

The results show that integration of long and short term model is beneficial to the response of the integrated system. This integration improve the system accuracy for both forecast horizon and also turns the resulting model generic. That affirmation can be proved by the close results for the tree types of load consumption. A generic forecasting system has a important advantage for commercial usage, because they could forecast many instances with only one model.

Finally, the main contribution of this work is a new neural model for load forecasting, by the topological level integration usage. With this integration, the computational system has proved flexible and capable to generating excellent results. Some other aspects of the load forecast in electric systems, like the expansion of the time horizon, will be published in future works.

REFERENCES

Abdel-Aal, R. E., Short-Term *Hourly Load Forecasting Using Abductive Networks*, 2004. IEEE Transactions on Power Systems, Vol. 19, No. 1.

Ardil, C., Manjunath, T., Mehta H., Mehta, A., 2007. A *Multi-Layer Artificial Neural Network Architecture Design for Load Forecasting in Power Systems*. International Journal of Applied Mathematics and Computer Sciences, Vol. 4, No. 4.

Botha E. C., Vermaak, J., 1998. *Recurrent Neural Networks for Short-Term Load Forecasting*. IEEE Transactions on Power Systems, Vol. 13, No. 1.

Charytoniuk, W.; Chen, M. S.; *Very Short-Term Load Forecasting Using Artificial Neural Networks*. IEEE Transactions on Power Systems, Vol. 15, No. 1, February 2000.

Drezga, I., Rahman, S.; *Short-Term Load Forecasting with Local ANN Predictors*. IEEE Transactions on Power Systems, Vol. 14, No. 3, August 1999.

Fukuyama,Y.; Matsui, T.; Iizaka, T.; *A Novel Daily Peak Load Forecasting Method using Analyzable Structured Neural Network*. IEEE T&D Asia, October 2002.

Funabashi, T.; Uezato, K.; Takara, H.; Senju, T.; *One-Hour-Ahead Load Forecasting Using Neural Network*. IEEE Transactions on Power Systems, Vol. 17, No. 1, February 2002.

Quintanilha, P., Souza, A., Leme, R., Carpinteiro, O., 2005. *A Hierarchical Hybrid Neural Model with Time Integrators in Long-Term Peak-Load Forecasting.* International Joint Conference on Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE. Vol. 5, Pages: 2960- 2965.

Saad, M.; Sood, V.; Kandil, N.; *Use of ANNs for Short-Term Load Forecasting*. IEEE Canadian Confrerence on Electrical and Computer Engineering, May 1999.

Seeger, M., Williams, C., Lawrence, N., 2003. *Fast Forward Selection to Speed Up Sparse Gaussian Process Regression*. Workshop on AI and Statistics.

Shirvany, Y., Hayati, M., 2007. *Artificial Neural Network Approach for Short Term Load Forecasting for Illam Region.* International Journal of Electrical, Computer, and Systems Engineering, Vol. 1 No. 2.

Swinder, D., Barth, R., Meibom, P., Weber, C., 2007. *Changes of System Operation Costs Due to Large-Scale Wind Integration*. Business and Policy Track: Integrating wind in electricity markets, 2007.

HYBRID DCA-PCA MULTIPLE FAULTS DIAGNOSIS METHOD

Funa Zhou[1,2], Tianhao Tang[2] and Chenglin Wen[3]
[1]*Computer&Information Engineering School, Henan University, Kaifeng, Henan, China*
[2]*Department of Electrical & Control Engineering, Shanghai Maritime University, Shanghai, China*
[3]*Institute of Information and Control, Hangzhou Dianzi University, Hangzhou, China*
Zhoufn2002@163.com, thtang@cen.shmtu.edu.cn, wencl@hdu.edu.cn

Keywords: Unknown fault pattern, Multiple faults, DCA, PCA, Fault diagnosis.

Abstract: As it can avoid the pattern compounding problem of PCA, designated component analysis (DCA) can be used to implement multiple fault diagnosis for a multivariate process. But designated fault pattern must be defined in advance, which limited its application in unknown fault diagnosis. In this paper, a hybrid DCA-PCA method is developed for unknown multiple faults diagnosis. the main idea is: Implement DCA in the first step. Removing the designated fault pattern from the observation data, then implement PCA to the residual, and use the first loading vector as the new fault pattern to extend the fault pattern base. In the third step, implement DCA for the new fault pattern and compute the significance of the new fault pattern. Simulation for data involved 4 faults shows the efficiency of the progressive DCA fault diagnosis method.

1 INTRODUCTION

Fault diagnosis is critical for large scale system since failure in a part may cause breakdown of the system or even disastrous accident (Zhou, 2000).

In general, fault diagnosis methods can be classified into 3 classes: quantitative model-based method, qualitative model-based method and process history based method, also called data driven method (Venkat,2003, Wen, 2008, Ku, 1995).

With the widely application of DCS and intelligent instrument in industry field, it is convenient to acquire and store a large amount of data on system operation. Since these data isn't efficiently used in monitoring, it is not surprise to face "data rich, information poor" problem. People are now realizing the significance of data driven monitoring method (Venkat, 2003, Yue, 2001).

Common used data driven diagnosis method includes: expert system method, ANN based method and statistical method (Venkat, 2003, He, 2007). Among data driven methods, statistical method seems to have been well studied and applied. And PCA/PLS based methods are the dominant ones. These PCA based methods are efficient in abnormal detection. But pattern compounding effect of PCA makes it unavailable to fault pattern recognition, especially for multiple faults diagnosis (Liu, 2002).

DCA is also a multivariate statistical information extraction method. It can avoid the pattern compounding problem of PCA, thus can be used to diagnose multiple faults (Liu, 2002, Zhou, 2009).

But

1) DCA requires all the designated patterns are orthogonal, which is impractical in most application;

2) DCA diagnosis method is validated only for those known fault patterns defined in advance.

The first problem has been solved in (Zhou, 2009). This paper focuses on developing a hybrid DCA-PCA method for unknown fault diagnosis.

2 PCA BASED FAULT DIAGNOSIS

The essence of PCA is a linear transform

$$v_i = b_i^T y \tag{1}$$

Where principal component v_i is the projection of observation variable $y = [y_1, y_2, \cdots, y_p]^T \in R^{p\times 1}$ on loading vector $b_i = [b_{i1}, b_{i2}, \cdots, b_{ip}]^T \in R^{p\times 1}$, which is the ith eigenvector of y 's covariance matrix Σ_y. For a sample size of n, equa. (1) expands into the following matrix form

$$V = B^T Y \tag{2}$$

where $Y \in R^{p\times n}$ is the observation matrix, $V \in R^{p\times n}$ is the scoring matrix.

PCA decompose the observation matrix Y as a sum of p matrices of rank 1

$$Y = \sum_{i=1}^{m} b_i v_i^T + \sum_{i=m+1}^{p} b_i v_i^T \quad (3)$$

Where m is the number of key principal component selected, $E = \sum_{i=m+1}^{p} b_i v_i^T$ is the residual.

Implement abnormal detection via the statistics T^2 and SPE (MacGregor, 1995, Zhang,2000). But they can not correctly recognize fault pattern.

3 DCA BASED FAULT DIAGNOSIS

The designated patterns are defined as $d_i = [d_{i1}, d_{i2}, \cdots, d_{ip}]^T$, where d_{ij} is 0 or 1 determined by the relation between fault and its symptom(Zhou,2009, Liu, 2004). Then, project y on d_i to get the designated components

$$w_i = d_i^T y \quad (4)$$

For a sample size of n

$$W = D^T Y \quad (5)$$

Thus Y can also be expressed as sum of p matrices of rank 1

$$Y = \sum_{i=1}^{p} d_i w_i \quad (6)$$

If there are only $l \le p$ variation pattern is designated, then (Zhou, 2009)

$$Y = \sum_{i=1}^{l} d_i w_i + E \quad (7)$$

Convergence of (7) has been proved in (Zhou, 2009).

Compute the significance of every designated pattern to determine whether the fault has occurred

$$P_i\% = \mathrm{var}(w_i) / trace(\Sigma_y) \quad (8)$$

But DCA is invalidated for unknown faults diagnosis. A hybrid DCA-PCA method will be developed to solve this problem.

4 HYBRID DCA-PCA MULTIPLE FAULT DIAGNOSIS METHOD

As it is known to all, PCA is a complete data-driven method. Although fault pattern PCA revealed makes no physical sense, it can determine a significant variation pattern of the abnormal system without any prior information. In this section, we develop a hybrid DCA-PCA multiple fault diagnosis method for the case when unknown new fault occurs.

First, define l $(l \le p)$ designated pattern as in (Zhou,2009), and implement DCA to the observation data; Then, remove the designated patterns defined in advance from the observation data to get the residual E ; Determine whether new fault is comprised in the residual according to the energy significance of the residual defined as $\|E\|_F$. $E = Y - \sum_{i=1}^{l} d_i w_i$ is large in the sense $\|E\|_F > \delta$ means that new fault occurred.

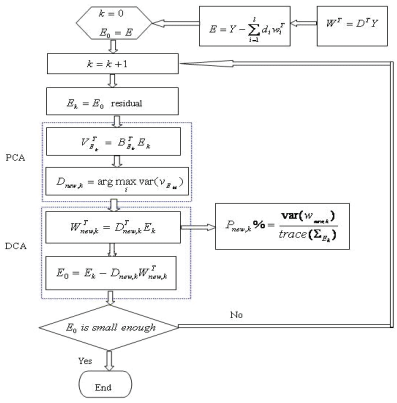


Figure 1: Hybrid DCA-PCA diagnosis method.

Implement PCA to E , and select the first loading vector as new fault pattern $D_{new,1}$. Another round of DCA is carried out for $D_{new,1}$. Repeating this process until residual is insignificant.

$$W^T = D^T Y \quad (9)$$

$$E = Y - \sum_{i=1}^{l} d_i w_i \quad (10)$$

$$V_E = B_E^T E \quad (11)$$

$$D_{new,1} = \max_i \mathrm{var}(v_{E_i}) \quad (12)$$

$$E_1 = E \quad (13)$$

$$W_{new,1} = D_{new1,1}^T E_1 \quad (14)$$

$$E_2 = E_1 - D_{new,1} W_{new,1} \quad (15)$$

$$V_{E_2} = B_{E_2}^T E_2 \quad (16)$$

$$D_{new,2} = \max_i \mathrm{var}(v_{E2_i}) \quad (17)$$

$$\vdots$$

Figure 1 depicts the hybrid DCA-PCA multiple fault diagnosis process.

5 SIMULATIONS

Simulations parameters used are: $p=15$, $n=1000$, $l=6$.

The observation data is generated by the composition of 10 variation patterns

$$Y=\sum_{i=1}^{10} d_i \bar{w}_i \tag{18}$$

Where $\bar{w}_i \sim N(0,\sigma_i^2)$ is the simulated designated component, $d_i \in R^{p\times 1}$ $(i=1,2,\cdots,10)$ are the 10 variation patterns to generate observation data Y. figure 2 depicts the contribution of each d_i to Y.

The first 6 pattern $d_1,\cdots,d_6$ are the designated pattern we selected, d_1,d_3,d_5 is the fault pattern, d_{10} is a fault pattern unconsidered in advance.

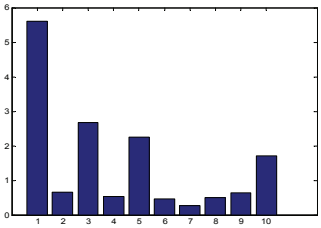


Figure 2: DC values for Generating observation.

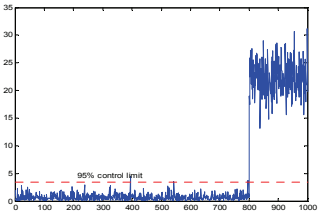


Figure 3: SPE chart for PCA.

For normal observation, $\bar{w}_i$ can be generated in MATLAB using "randn" and some linear operation

For fault case, d_1,d_3,d_5,d_{10} are amplified from sampling time 801 to 1000

$$\bar{w}_i=\bar{w}_i+5\,\mathrm{var}(\bar{w}_i)\quad i=1,3,5,10 \tag{19}$$

5.1 PCA based Fault Diagnosis

The SPE chart of PCA are shown in figure 3. Figure 3 indicates that system considered is abnormal from 801st sample point. However, SPE chart can not tell what faults occur.

5.2 Hybrid DCA-PCA Multiple Faults Diagnosis

Implement DCA to the observation, and illustrate the significance of each designated pattern in table 2. From table 2, we can see that, the fist, the third and the fifth variation pattern are the 3 significant ones of the 6 designated patterns. According to the significance of the observation to each designated, we can conclude that faults corresponding to these 3 designated patterns have occurred in the system.

Table 2: Significance of the designated pattern in D.

	d_1	d_2	d_3	d_4	d_5	d_6
d_i%	0.4123	0.0368	0.1946	0.0359	0.1839	0.0378

Figure 6 draws the Shewhart chart of every designated component in. It indicates that the 1st, 3rd, and 5th designated component's Shewhart chart exceed the control limit from 801 to 1000. Figure 6, confirms that faults corresponding to the 1st, 3rd, and 5th designated patterns have occurred in the system.

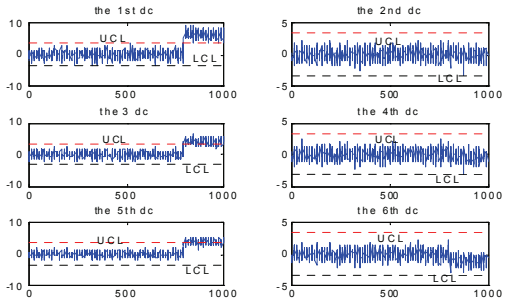


Figure 6: Shewhart chart for 6 DC.

For the case $p=15$, $n=1000$ and $l=6$, statistical result of more than 100 times simulation shows that the threshold is reasonable

$$\delta=10 \tag{20}$$

Removing the 6 designated variation pattern to get the residual E_0. The norm of the residual is

$$\|E_0\|_F=12.7524>\delta \tag{21}$$

It is possible that at least one new fault is still included in the residual. Implement PCA to the residual E_0, and take the first load vector as a new

fault pattern. Then implement DCA to E_0 for d_{new1}. And compute its significance

$$d_{new1}\% = 0.6083 \quad (22)$$

The Shewhart chart of this new designated component is depicted in figure 7. Figure 7 tells us that fault corresponding to d_{new1} has occurred in the system. Removing the new fault pattern d_{new1} from E_0 we have the residual of this DCA step

$$\|E_1\|_F = 7.2042 \quad (23)$$

To the residual E_1, Shewhart chart for the secondnew designated component, figure 8 is within the control limit, which will confirm that $\delta = 10$ is reasonable

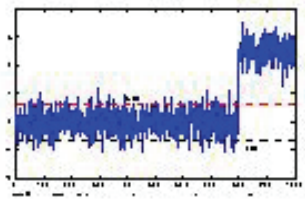


Figure 7: Shewhart chart for the 1st new dc.

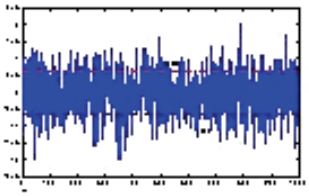


Figure 8: Shewhart chart for the 2nd new dc.

From the above simulation research, we can conclude that d_1, d_3, d_5 and d_{10} occurred in the system. This is basically the same as the simulation manner that we used to generate Y.

6 CONCLUSIONS

DCA can avoid pattern compounding problem of PCA. But it is invalidated for unknown faults diagnosis. In this paper, a hybrid DCA-PCA method for unknown multiple fault diagnosis.

Some data driven methods other than PCA can be used to the residual to estimate the new fault pattern to make it physical sense.

ACKNOWLEDGEMENTS

This paper is supported by NSFC (60804026); International cooperation project of Zhejiang (2006C24G2040012), Natural science fund of Henan (2009A510001) International cooperation project of Henan (094300510043), Key disciplines of Shanghai Municipality (J50602), Development Project (08YZ109) from Shanghai Municipal Education Commission.

REFERENCES

Donghua Zhou, Yinzhong Ye, 2000. Modern fault diagnosis and tolerant control[M], Beijing, Qstinghua Publishing House (in Chinese).

Venkat Venkatasubramanian, Raghunathan Rengaswamy, Kewn Yin, Surya N. Kavuri,2003. A review of process fault detection and diagnosis Part I[J]: quantitative model-based methods. Computers and Chemical Engineering 27 (2003):pp293-311.

Qingbo He, 2007. Application Multivariate statistical analysis in machine state monitoring and diagnosis [D], PHD thesis, University of Science and Technology of China (in Chinese).

Yue H H, Qin S J, 2001. Reconstruction based fault identification using a combined index. Industrial and Engineering Chemistry Research[J], 40(20): 4403-4414

J.F. MacGregor and T. Kourtl, 1995. Statistical process control of multivariate processes[J], Control Fag. Practice, VoL 3, No. 3, pp. 403-414.

Yegang Liu, 2002. Statistical control of multivariate processes with applications to automobile body assembly (D). PHD, University of Michigan.

Jie Zhang, Xianhui Yang. Multivariate statistical control[M], Beijing, Chemistry Industry Publishing House, 2000 (in Chinese).

Funa Zhou, Chenglin Wen, Tianhao Tang, 2009. DCA based multiple faults diagnosis method, accepted by ACTA AUTOMATICA SINICA (in Chinese).

Chenglin Wen, Jing Hu, Tianzhen Wang, Zhiguo Chen. RPCA and it's application in data compression and fault diagnosis, ACTA AUTOMATICA SINICA 34(9) : 1128-1139 (in Chinese).

Ku, W., Storer, R.H., and Georgakis,C, 1995. Disturbance detection and isolation by dynamic principal component analysis [J], Chemometrics and Intelligent Laboratory Systems, 30:179-196.

DETECTION OF A FAULT BY SPC AND IDENTIFICATION

A Method for Detecting Faults of a Process Controlled by SPC

Massimo Donnoli
DSEA – Dept. Electrical Systems and Automation, University of Pisa, Italy
m.donnoli@lucchini.com

Keywords: Statistical process control, Multivariate Hotelling statistic, System identification.

Abstract: A method for detecting the nature of a fault of a process controlled by SPC (Statistical Process Control) is presented. The method use the integration of SPC , traditional APC (Automatic Process Control) and the System Identification technique . By a statistical on line control of the parameters of a transfer function and the identification of the transfer function itself, the case of a fault due to a change in the system is recognised. An algorithm called 'batch control' for the implementation of the method in a real plant is proposed.

1 INTRODUCTION

The objective of Statistical Process Control (SPC) is to detect situation of change of the natural behaviour of a process by monitoring on line the key product variables and detecting the cause of the fault, indicating which variable or group of variables contributes to the signal.

A lot of technique has been developed especially due to the large and different areas where the SPC could be applied.

If, traditionally, the SPC has been developed especially to monitor the complicated processes of chemical plants, after that, the big growth of the information technology in the industries and the large amount of process measures collected in the data base of the control systems, has allowed the implementation of SPC on almost every kind of plant. By the way the common goal for the most application is still to monitor the quality of the process, treating the manufacturing process itself as a black box, of which we know the inputs and outputs, ignoring the others information of the nature of the process.

In fact traditionally SPC and APC (Automatic Process Control) have been developed in parallel and only in the last years there has been works where researchers have made the integration of the two areas (Tsung,1999).

Another point to remark is that the traditional SPC approach, that is still the most diffused in many kind of industries, is essentially the univariable SPC: by the implementation of control charts like Shewhart, Cusum, etc.. we look the magnitude of the deviation of each variable independently of all others as they are perfectly independent in the process.

But the being 'in control' of a process is essentially a multivariable property : in the modern industrial processes the variables are non independent of one another and only if the simultaneous state of them all is in the joint confidence region defined for the system, we could say that the system is in control : by examining one variable at time with the traditional charts it could be that every variable is in the correct range but the common state is not (Kourti & MacGregor,1994).

For this have been developed multivariable methods that can treat all the variables simultaneously.

The principal is the Hotelling or T^2 statistic : it transforms the state of all the variables in the calculation of the value of a single variable which can be monitor for detecting faults.

If this is a great effort to solve the problem in a very useful way, on the other hand we have now the problem to detect the cause of the fault, the variable responsible.

This paper is organized as follows: in the next section the basic concepts of SPC multivariable, the Hotelling statistic and the interpretation of a T^2 value are recalled. In sections 3 and 4 the main advantages of SPC - APC integration and SPC - System Identification integration are presented.

In section 5 a method for statistically monitoring the points of a system transfer function is presented. In section 6 are given the simulation results for a typical industrial APC. In section 7 an algorithm called 'batch control' for the implementation of the technique in some kind of industrial processes is presented . Finally some conclusions are given.

2 SPC MULTIVARIABLE

Suppose that our process has 2 measures represented by 2 random variables x1,x2 uncorrelated, with mean value $\mu 1, \mu 2$ and variances $\sigma 1, \sigma 2$ respectively. Consider the distance of an observation point from the mean point in the plane (x1,x2). Instead of the usual Euclidean distance we consider the relationship :

$$\frac{(X1-\mu 1)^2}{\sigma 1^2}+\frac{(X2-\mu 2)^2}{\sigma 2^2}=(SD)^2 \quad (1)$$

This is called 'statistical distance' (SD) . For an observation the contribution of each coordinate to determining the distance is weighted inversely by its standard deviation, that means that a change in a variable with a small standard deviation will contribute more to the statistical distance than a change in a variable with a large standard deviation. It follows that the statistical distance is a measure of the respect of the statistical behaviour of the two variables.

If they are correlated it will be an additional term in x1 x2 and in the plane the curve with constant SD will be an ellipse, eventually tilted according to the correlation sign.

In general if we have a process of p variables and we have n observation vector of the p variables with the system 'in control' (or what is called the Historical Data Set (HDS) of the system) we can calculate an estimation of the main vector and of the covariance matrix of the random variables :

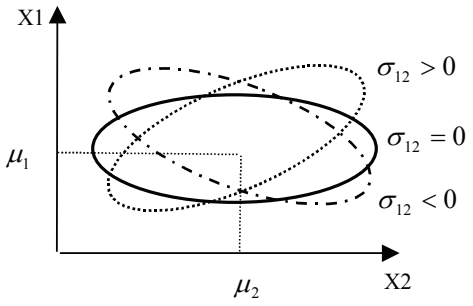


Figure 1: Curves with constant SD for two variables.

$$\mu=\frac{1}{n}\sum_{i=1}^{n}X_i \quad (2)$$

$$\Sigma=\begin{bmatrix}\sigma 11 & \sigma 12 & \cdots & \sigma 1p\\ \sigma 21 & \sigma 22 & \cdots & \sigma 2p\\ \vdots & \vdots & \cdots & \vdots\\ \sigma p1 & \sigma p2 & \cdots & \sigma pp\end{bmatrix} \quad (3)$$

For a generic observation we can then compute the quantity :

$$T^2=(X_i-\mu)^T\,\Sigma^{-1}(X_i-\mu) \quad (4)$$

This is an univariate quantity that is called Hotelling Statistic or T^2 . It is clear that this is the multivariable generalization of the statistic distance, in words a measure of the closeness (in a statistical way) of the observation to the behaviour of the system expressed by the HDS .

The curves with constant T^2 are then hyper-ellipsoids in the p dimensional space.

Considering T^2 like a random variable we can see that , in the case of μ and $\sum$ estimate by the observations, it follows the distribution of an F random variable of p, n-p degrees of freedom.

Let α be the first type error and let $F_{(\alpha,p,n-p)}$ be the value f of F | P(F<=f) = 1-α (P : probability of) we can then calculate an Upper Control Limit (UCL) for T^2 :

$$UCL=\frac{p(n+1)(n-1)}{n(n-p)}F_{(\alpha,p,n-p)} \quad (5)$$

We can say that if we are under the UCL we have a probability 1- α to say that the system is in control when really it is .We can see that choosing an α smaller led to have a second type error β bigger, that is a greater probability to not detect a fault when it really exists.

In the industrial processes both errors are important: α is the representation of the false alarms that can led to stop the production in vain, while β, if large, can led to not detect situations of real out of control.

Generally β is set low because is preferable to have some false alarm than to not detect a fault .

In our examples we have chosen $\alpha = 0.1$.

2.1 Interpretation of T^2 Signals

The T^2 converts a multivariable problem to the calculation of an univariate quantity. But signal interpretation requires a procedure for isolating the responsible of the fault because the contribution could be attributed to individual variables being

outside their allowable range of operation or to a fouled relationship between two or more variables.

Several solutions have been presented for the problem of interpreting a multivariate signal.

One that we show here for example is the MYT decomposition (Mason –Young 2002), that uses an orthogonal transformation to express the T^2 values as two orthogonal equally weighted terms :

$$T^2 = T_1^2 + T_{2,1}^2 \quad (6)$$

$$T_1^2 = (x_1 - \overline{x_1})^2 / \sigma_1^2 \quad (7)$$

$$T_{2,1}^2 = (x_2 - \overline{x_{2,1}})^2 / \sigma_{2,1}^2 \quad (8)$$

where x2,1 is the estimate of the conditional main of x2 for a given value of x1 and $\sigma 2,1$ is the corresponding estimate of the conditional variance of x2 for a given value of x1. A large value of the first term (called 'unconditional term') implies that the observed value of the variable is outside his operational range as was on HDS, while a large values of the second term ('conditional term') implies that the observed value of one variable is not where it should be relative to the observed value of the others variables.

By subsequently eliminations of the unconditional terms that signal and iterative decomposition of the conditional terms that signal, it is possible to isolate the variable or group of variables responsible of the fault. We can say that all this methods have in common iterative procedures and sometimes great computational efforts to reach the scope.

3 SPC APC INTEGRATION

Combining SPC with APC could be an improvement of the global control of the system.

The scheme is the following :

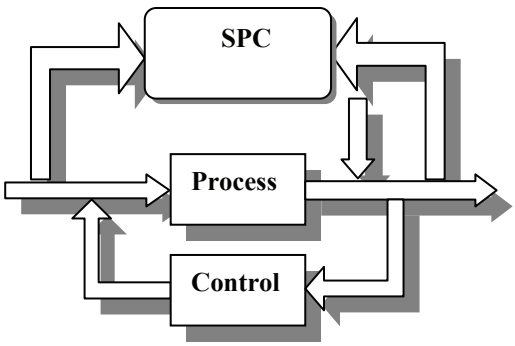


Figure 2: SPC - APC integration.

There is an interaction between SPC and APC : SPC controls not only the variables in and out of the process but also the signals of the control: the result is that the process is no more treated like a black box but the information in the APC are used.

Our procedure uses the information of the transfer function of the system: the change of the system is statistically monitored controlling the parameters of the transfer function

4 SPC – SYSTEM IDENTIFICATION

An industrial system is typically formed by automation systems that process the product .

We can divide the entire set of variables of the manufacturing process into 'process variables', meaning that they are measures made above the product, (temperature, time of operations etc) and 'control variables' (like input - output of the motors, signals of the electronic devices , etc..) that reflect the behaviour of the automation systems.

We propose the identification of the transfer functions of the control systems and the statistical monitoring of the transfer functions to detect deviation of the automation system from their normal behaviour .

The identification at the same time allow a better design of the control and a more easily detection of the fault due to the automation systems and not to the process .

5 STATISTICAL CONTROL OF A TRANSFER FUNCTION

Consider the simple transfer function of a system G:

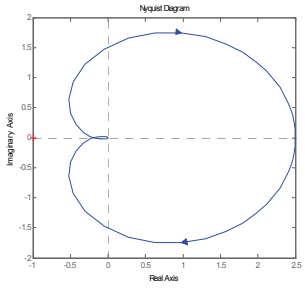


Figure 3: Nyquist plot of a transfer function G.

We can consider the transfer function as a function of 3 parameters : amplitude, phase and

frequency. So we can plot the function in the 3d space of that measures :

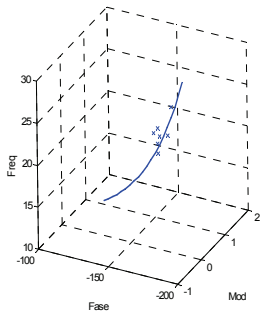


Figure 4: Observations of G points in the space amplitude-phase-frequency.

If we can do measurements of that parameters our observations will be point around the real curve of the transfer function as shown above.

If we plot the curve at constant statistical distance they will be ellipsoid in the 3d space with their major axis tangent at the curve of the G in the point we are sampling it. That tangent is the best linear approximation of the population, because the variables follow the relationship of the G.

So we have that schema for the identification and statistic control of the G:

Make an HDS by collecting observations of the points of the transfer function while the system is in his normal behaviour.

Estimate from the HDS the covariance matrix and the average value of the observations (2) (3).

Set an UCL for the Hotelling statistic chosen a desired α (5).

Monitoring the T2 control chart build with (4).

6 SIMULATION RESULTS

We present the simulation result of a typical industrial APC :

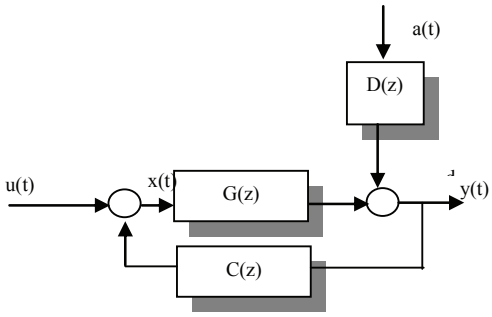


Figure 5: Model for simulations.

formed by a system controlled by a PID controller and disturbed with a coloured noise in the output, according to the typical modelling of an industrial noise.

$$x(t) = -k_P y(t) - k_I \sum_{j=0}^{\infty} y(t-j) - k_D (y(t) - y(t-1))$$

$$D_t = \frac{1-\theta Z^{-1}}{1-\phi Z^{-1}} a_t \qquad (9)$$

$$G(z) = \frac{0.9z}{z^2 - 1{,}2z + 0.32}$$

$$a_t \in N(0, \sigma_a)$$

For the on line measurement of the point of the G we have used the technique of the Descriptive Function by inserting a relay with hysteresis h and amplitude A in the control loop.

Leading the system in a controlled oscillation we can see that the relationship

$$G(\omega_c) = -\frac{1}{F} = \frac{\pi Y_c}{4A} e^{j(-\pi + arcsen\frac{h}{Y_c})} \qquad (10)$$

allow to measure a point of the G at the oscillating frequency in amplitude and phase. By varying the hysteresis it is possible to evaluate several points.

We show the result of 50 observations after a change of 10% of the first pole of the system

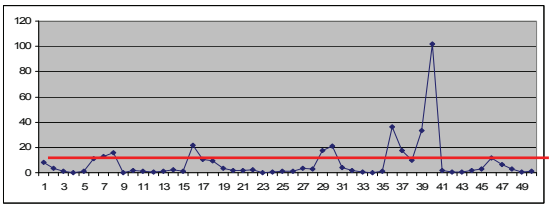


Figure 6: T^2 chart of 50 observation after a change of 10% of the first pole of the system.

The T^2 signals the change in the system with an ARL (Average Run Length) of 8.

7 PROPOSED BATCH CONTROL ALGORITHM

We have experienced several industrial process for which the production is in two phase : a phase of production , by the working of a material in input , and a phase of wait for the other material to come. In this case we propose to apply the SPC for the 'process variables' during the phase of the effective production (say 'batch on') and to take advantage of the waiting phase (say 'batch off') for the identification of the automation system and applying then the SPC to the 'control variables'. That schema

overcome the problem of detecting if the nature of a fault is in the process or in the automation systems, giving a separation of the two analysis at the origin.

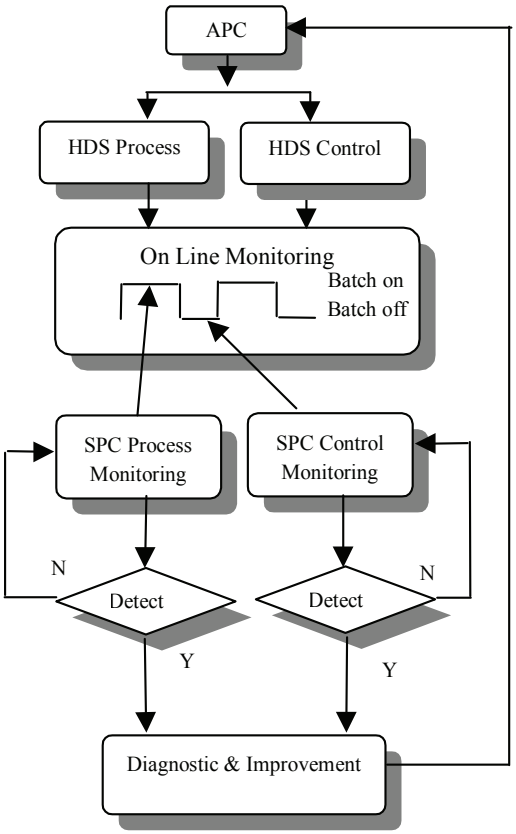


Figure 7: Algorithm of implementation.

When a signal of the control charts is detected we can say if it is due to the process or to a change of the transfer function of one of the automation systems in the manufacturing process. The phase of diagnostic and improvement is able then to correct the process variable signalling (for ex. adjusting a temperature) or operate directly on the APC that is controlling the automation system changed to compensate for the change, taking advantage of the identification made of the transfer function.

8 CONCLUSIONS

In this paper a method for monitoring on line the behaviour of an automation system is presented. The method has the advantage of using the identification of the transfer function of the system , so it can be used on the APC to compensate the changes, and with the advantage for the SPC to provide a direct separation of the possible causes of fault.

Results of simulations are given and finally an implementation schema of the method in a real process is proposed . For further researches we are applying the algorithm proposed in a rolling-mill plant for the production of railways, that has a production cycle on-off like the one described in the paper.

REFERENCES

Mason, R.L., Young, J.C., 2002. Multivariate Statistical Process Control with Industrial Application. ASA-SIAM.

Tsung, F., Apley, D.W., 2002. The dynamic T2 chart for monitoring feedback-controlled processes. *IIE Transactions (2002) 34,1043-1053* IIE.

Kourti, T., MacGregor, J.F., 1994. Process analysis, monitoring and diagnosis,using multivariate projection methods. *Chemometrics and Intelligent Laboratory Systems 28 (1995) 3-21*. ELSEVIER.

Tsung, F., 1999. Improving automatic-controlled process quality using adaptative principal component monitoring. *Qual.Rielab.Engng.Int. 15:135-142(1999)* John Wiley & Sons;Ltd.

Ljung, L., 1987. System Identification: Theory for the user. Prentice-Hall.

APPLYING SUB-POPULATION MEMETIC ALGORITHM FOR MULTI-OBJECTIVE SCHEDULING PROBLEMS

Yen-Wen Wang
Department of Industrial Engineering and Management, Chin Yun Tech. University
229 Chien-Hsin Rd., Taoyuan, Taiwan. R.O.C.
ywwang@cyu.edu.tw

Chin-Yuan Fan
Department of Industrial Engineering and Management
Yuan-Ze University, Taoyuan, Taiwan, R.O.C.
s948906@mail.yzu.edu.tw

Chen-Hao Liu
Department of Information Management, Kainan University
Taoyuan Taiwan, R.O.C.
chliu@mail.knu.edu.tw

Keywords: Flowshop scheduling problem, Multi-objective scheduling, Memetic Algorithm.

Abstract: Memetic Algorithm is a population-based approach for heuristic search in optimization problems. It has shown that this mechanic performs better than traditional Genetic Algorithms for some problem. In order to apply in the multi-objective problem, the basic local search heuristics are combined with crossover operator in the sub-population in this research. This approach proposed is named as Sub-population with Memetic Algorithm, which is applied to deal with multi-objective Flowshop Scheduling Problems. Besides, the Artificial Chromosome with probability matrix will be introduced when the algorithm evolves to certain iteration for injecting to individual to search better combination of chromosomes, this mechanism will make faster convergent time for evolving. Compares with other three algorithms which are MGISPGA, NSGA-II and SPEA2, the experiments result show that this algorithm possess fast convergence and average scatter of Pareto solutions simultaneously for solving multi-objective Flowshop Scheduling Problems in test instances.

1 INTRODUCTION

In the operations research literature, Flowshop scheduling is one of the most well-known problems in the area of scheduling. Flowshops are useful tools in modeling manufacturing processes. A permutation Flowshop is a job processing facility which consists of several machines and jobs to be processed on the machines. In a permutation Flowshop all jobs follow the same machine or processing order and job processing is not interrupted once started. Our objective is to find a sequence for the jobs so that the makespan or the completion time is a minimum.

In this research, we take a close look at the evolutionary process for a permutation Flowshop scheduling problems and come out with the new idea of generating artificial chromosomes to further improve the solution quality of the genetic algorithm. To generate artificial chromosomes, it depends on the probability of each job at a certain position. The idea is originated from Chang et al.(2005) which propose a methodology to improve Genetic Algorithms (GAs) by mining gene structures within a set of elite chromosomes generated in previous generations. Instead of replacing the crossover operator and mutation operator due to efficiency concern, the proposed algorithm is embedded into simple GA (SGA) and non-dominated sorting genetic algorithm-II (NSGA-II). The probability model acquired from the elite chromosomes will be integrated with the genetic

operators in generating artificial chromosomes, i.e., off-springs which can be applied to enhance the efficiency of the proposed algorithm. Apart from our previous researches, Harik (1999), Rastegar (2006), Zhang (2005) have discussed and proved the genetic algorithm which is based on the probability models. For a complete review of the relative algorithms discussed above, please refer to Larranaga (2001), Lozano (2006), and Pelikan (2002). In most recent works of evolutionary algorithm with probability models, they all concentrate on solving continue problems rather than discrete problems. There are only few researches in applying evolutionary algorithm with probability models to resolve discrete problems.

2 METHODOLOGY

A new approach is developed in this research which is called SPMA. The method is proposed to solve Flowshop scheduling problems and will be compared with SPGA, NSGA-II and SPEA-II. Through literature reviews, we find that SPGA has very good diffusivity when solving multi-objective problems; however, as for convergence, there still remains room for improvement. Thus, the research tries to strengthen the solution convergence of SPGA by mining gene structures and local search heuristic. Except for the original mining gene structures (Chang 2005), we called Artificail chromosomes (AC).

2.1 Generating Artificial Chromosomes

During the evolving process of the GA, all the chromosomes will converge slowly into certain distribution after the final runs. If we take a close look at the distribution of each gene in each assigned position, we will find out that most the genes will be converged into certain locations which means the gene can be allocated to the position if there is a probabilistic matrix to guide the assignment of each gene to each position. Artificial Chromosomes (AC) are developed according to this observation and a dominance matrix will record this gene distribution information. The dominance matrix is transformed into a probability matrix to decide the next assignment of a gene to a position. Consequently, AC is integrated into the procedure of genetic algorithm and it attends to improve the performance of genetic algorithm. The primary procedure is to collect gene information first and to use the gene information to generate artificial chromosomes. Before collecting the gene information, AC collects the chromosomes whose fitness is better by comparing the fitness value of each chromosome with average fitness value of current population. Then artificial chromosome is embedded into the genetic algorithm. The detailed steps are described in the following:

1. To convert gene information into dominance matrix:

   Before we collect gene information, selection procedure is performed to select a set of chromosomes. Then, for a selected chromosome, if job i exists at position j, the frequency is added by 1. To demonstrate the working theory of the artificial chromosome generation procedure, a 5-job problem is illustrated. Suppose there are ten sequences (chromosomes) whose fitness is better than average fitness. Then, we accumulate the gene information from these ten chromosomes to form a dominance matrix. As shown in the left-hand side of Figure 1, there are two job 1, two job 2, 2 two 3, one job 4, and three job 5 on position 1. Again, there are 3 job 1, 1 job2, 2 job3, 3 job4, and 1 job5 on position 2. The procedure will repeat for the rest of the position. Finally, the dominance matrix contains the gene information from better chromosomes is illustrated in the right-hand side of Figure 1.
2. Generate artificial chromosomes:

   As soon as we collect gene information into dominance matrix, we are going to assign jobs onto the positions of each artificial chromosome. The assignment sequence for every position is assigned randomly, which is able to diversify the artificial chromosomes. After we determine the assignment sequence, we select one job assigned to each position by roulette wheel selection method based on the probability of each job on this position. After we assign one job to a position, the job and position in the dominance matrix are removed. Then, the procedure continues to select the next job until all jobs are assigned. Assume the first job is to be assigned at position 3 in the beginning. The frequency of each job at position 3 is [1, 3, 1, 1, and 4] starting from job 1 to job 5. Because the number of total frequency is 10, the corresponding probability for job 1 is 1/10; job 2 is 3/10, and so on. Then, we accumulate the probability from job 1 to 5 and roulette wheel select is able to apply this accumulated probability. If a random probability 0.6 is generated, then job 4 is assigned to position 3.
3. Replacement strategy:

After embedding artificial chromosomes into the population, we use l + k strategy, which combines previous parent population and artificial chromosomes. Then, we select better l chromosomes from the combined population. Consequently, better solutions are preserved to the next generation.

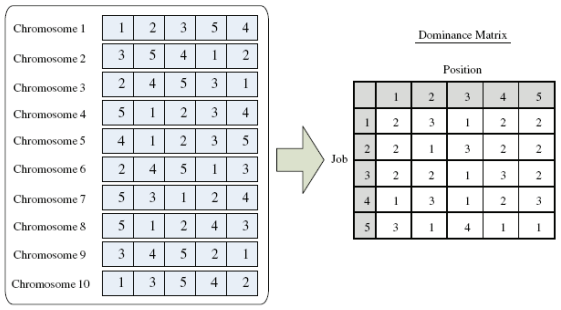


Figure 1: To collect gene information and converted into a dominance matrix.

During the assignment of each job to a specific position, the dominance matrix will be updated continuously. For example, after assigning job 4 at position 3 and suppose position 2 is the next one to be assigned. An updated dominance matrix is shown in Figure 2.

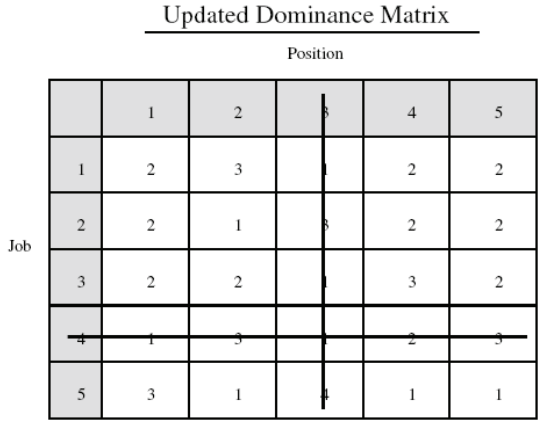


Figure 2: The updated dominance matrix after assigning job 4 at position 3.

Next, the probability of each job is recalculated as well as the accumulated probability. Then, a roulette wheel selection method will select a job based on the probability of each job. Consequently, the algorithm iteratively assigns jobs to vacant positions until all jobs are assigned.

2.2 The procedure of SPMA

The detailed procedure of SPMA is shown as the following:

(Ns is the number of sub-populations; g is the number of generations; k is the interval number of artificial chromosomes' generations.)

```
1.  Initialize();
2.  DividePopulation();
3.  AssignWeightToEachSubGroup();
4.  for i=1 to Iterations
5.    for j=1 to Groups
6.      if i % k == 0
7.        ArtificialChromosome(j);
8.  <GenerateArtificialChromosome(j);
9.         LocalSearch(j);
10.        SurvivalofThefittest(j);>
11.     else
12.       GeneAlgorithm(j);
13.       <Selection(j);
14.        Crossover(j);
15.        Mutation(j);
16.        LocalSearch(j);
17.        SurvivalofThefittest (j);>
18. FindPareto(Groups);
19. UpdatePareto(Groups);
```

Compared with SPGA, this approach is different in that it has the mechanism of creating AC, local search heuristic and the sorting information of chromosomes in each mutation is recorded for the use of creating AC and placing them in the mating pool for evolution. In the end, $D1_R$ is a metric which considers the convergence and diversity simultaneously (Knowles, 2002) and it is adopted in this research to evaluate the solution quality. Its main concept is to calculate the shortest distance between each solution in the Pareto-Solution set and the set to be compared with, and then calculate the mean value. The smaller $D1_R$ is the better.

3 EXPERIMENTAL TESTS

The research uses the Flowshop scheduling case study by Ishibuchi (2003) in which four types were included in the bi-objective flow-shop problems; they were 20 jobs in 20 machines, 40 jobs in 20 machines, 60 jobs in 20 machines and 80 jobs in 20 machines. Two objectives are the total completion time (Cmax) and maximum tardiness (Tmax). The processing and completion time are the same as used in Ishibuchi et al. (2003). The experimental results will be compared with those of SPGA, NSGA-II and SPEA-II. The testing result of this sample problem is depicted in Table 1 and Figure 3. It is obvious that SPMA performs better in the small and medium size problem (job = 20, 40, 60). And it deserve to be

mentioned is the Std. of SPMA is much less than other models.

Table 1: The algorithm comparison with other methods.

Instance (job)	NSGA II		SPEA II	
	Ave.	Std.	Ave.	Std.
20	43.05	14.42	37.35	14.22
40	146.36	28.71	138.6*	19.9
60	321.12	57.86	291.02	52.78
80	424.96	93.92	394.14	63.79
Instance (job)	SPGA		SPMA	
	Ave.	Std.	Ave.	Std.
20	38.62	9.44	22.46*	8.89
40	146.21	21.09	142.40	20.03
60	341.86	94.15	261.24*	34.49
80	344.45*	99.12	515.34	75.72

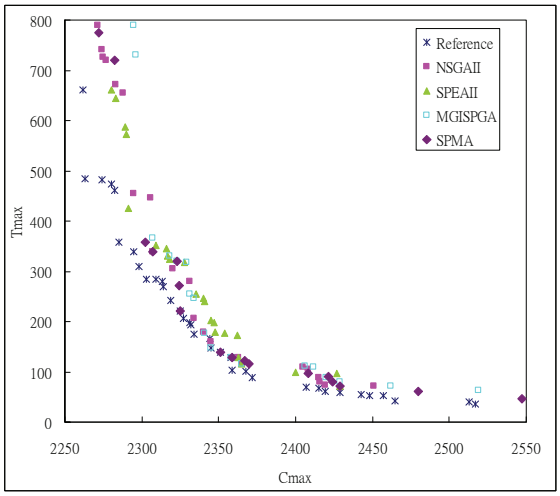


Figure 3: The plot of algorithms with reference Pareto set of SPMA.

According to the above-mentioned four testing results, we find that when solving more complex problems, it is harder to find the improving effectiveness of SPMA. Along with the increasing number of jobs, the problems become more complex and thus the improving effectiveness of SPMA can't be obviously noticed.

4 CONCLUSIONS

Through this study, we can verify that by combining AC and local search heuristic with SPGA, multi-objective scheduling problems can be solved more effectively, especially in the small and medium size problem. In the future, SPMA can be further extended to three objectives or multidimensional continuous problems. And the procedures of SPMA might be improved to deal with large size problem.

Further investigation will be carried out to examine whether it is possible to generate elite chromosomes through better mining algorithms. It is also suggested that different objectives of Flowshop scheduling problems can be further tested such as the arrival time of job is considered, and those with more complex requirements such as sequence dependent setup times.

REFERENCES

Chang, P. C., Chen, S. H., and Lin, K. L., 2005, Two phase subpopulation genetic algorithm for parallel machine scheduling problem, *Expert Systems with Applications*, 29(3), 705–712.

Harik, G.R., Lobo, F.G., and Goldberg, D.E., 1999, The compact genetic algorithm, *IEEE Transactions of Evolution Computing*, 3 (4), 287 – 297.

Ishibuchi, H., Yoshida, T. and Murata, T., 2003, Balance between Genetic Search and Local Search in Memetic Algorithms for Multi-objective Permutation Flowshop Scheduling, *IEEE Trans on Evolutionary Computation*, 7(2), 204-223.

Larranaga, P., and Lozano, J.A., 2001, Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation, *Kluwer, Norwell.*

Lozano, J.A., Larranaga, P., Inza, I., and Bengoetxea, E., 2006, Towards a New Evolutionary Computation, *Springer.*

Pelikan, M., Goldberg, D.E., and Lobo, F.G., 2002, A survey of optimization by building and using probabilistic models, *Computational Optimization and Applications,* 21 (1), 5 – 20.

Rastegar, R., and Hariri, A., 2006, A step forward in studying the compact genetic algorithm, *IEEE Transactions of Evolution Computing*, 14 (3), 277 – 289.

Zhang, Q., Sun, J., and Tsang, E., 2005, An evolutionary algorithm with guided mutation for the maximum clique problem, *Evolutionary of Computing*, 9 (2), 192 – 200.

RELATIONSHIPS BETWEEN BATCH SIZES, SEQUENCING AND LEAD-TIMES

Vladimír Modrák
Faculty of Manufacturing Technology, Technical University of Košice, Bayerova 1, Prešov, Slovakia
vladimir.modrak@tuke.sk

Ján Modrák
T-Systems Slovakia s.r.o.. Košťova 1, Košice, Slovakia
jan.modrak@t-systems.sk

Keywords: Simulation, Manufacturing lead time, Sequencing, Batch production.

Abstract: This paper treats the optimization of production batches by computer simulation in a manufacturing company producing electric and pneumatic actuators. In its introduction part the article deals about a wider context of batch production optimization. Subsequently, the paper presents a procedure for creation of a simulation model in SIMPLE ++ software environment. Based on simulation of a manufacturing process, selected dependences of lead manufacturing time on changes of production sizes was studied. As a result of optimization has been determined optimal minimal value of production sizes, by which minimal lead-time can be achieved.

1 INTRODUCTION

Manufacturing lead time reduction is one of the most critical issues in gaining a competitive advantage in the marketplace. Manufacturing lead time (MLT) can be defined as the time span from material availability at the first processing operation to completion at the last operation. Obviously, there are abundant reasons to reduce lead times in most organizations. Obviously, reducing MLT doesn't mean speeding up operation times, but all efforts should be focused on shortening changeover times, eliminating needless operations and reduction of production and logistical bottlenecks. Especially, batch sizes effect on MLT through changeover times. When using larger batches, then changeover times compared with the manufacturing times will be insignificant. Contrariwise, if applying smaller batches, then longer changeover times would reduce the capacity of the factory greatly. Research in this paper is oriented on the optimization of production batches by computer simulation in a manufacturing company, in which mentioned issues present a topical problem. The paper is structured as follows. After a short section on related work, the theoretical background is outlined. Then, testing of relations between batch sizes, sequencing and lead times is treated. Finally, discussion on obtained results is presented.

2 RELATED WORK

Importance of manufacturing lead times in generally depends on production policies. Manufacturing policy in this relation is associated with one of the two strategies: Make-to-Order (MTO) or Make-to-Stock (MTS). In a case of MTO, some products are commonly under extreme pressure, which creates a situation where certain products need to get priority over other products (Akkerman and van Donk, 2007). However, this prioritization doesn't solve problem with excessive throughput times in the plants. Thus, the same authors used the average lead time to investigate the effects of different product mixes. Fahimnia et al. (2009) analyzed obstacles in reducing manufacturing lead times and observed that relatively long MLT is the major cause of inefficient manufacturing, since it generates large amount of wastes and creates considerable environmental encumbrance. In a context of integrated supply chain, duration of lead time of original equipment manufacturers causes a different retailer's profit.

Based on this assumption Mukhopadhyay (2008) studied optimal policies of retailers in different cases depending on the contract type with original equipment manufacturers. Guiffrida and Nagi (2006) focused their research on strategies for improving delivery performance in a serial supply chain based on evaluation of delivery performance. By them, 'delivery performance is classified as a strategic level supply chain performance measure'. Instructive consequences formulated Wacker (1996): 'If customer lead time is longer than manufacturing lead time, firms deliver from their production system and if customer lead time is shorter than manufacturing lead time, firms store finished goods inventory and incur holding costs'. A number of other authors studied the relationship between batch sizes and length of lead time. For instance, Kuik and Tielemans investigated the relationship between batch sizes and lead-time variability, or Millar and Yang (1996) analyzed relations between batch sizing and lead-time performance through the use of a queuing network model. Summarily stated, there are many options to achieve lead-time reduction.

3 THEORETICAL POSITION

In calculating the manufacturing lead time, the structure of the activities in production is one of decisive issue. Groower (1987) proposed to divide main production activities in two main categories, operation and no operation elements, excluding setup procedures that are generally required to prepare each production machine for the particular product. Thus, MLT is calculated as the sum of setup time, processing time, and non-operation time (Groover, 1987):

$$MLT = \sum_{i=1}^{n_m} (T_{sui} + QT_{oi} + T_{noi}) \quad (1)$$

where i indicates the operation sequence in the processing and $i = 1,2,\ldots,n_m.$; T_{su} represents setup time for each process; T_o is operation (processing) time per item per process; Q demonstrates batch size and finally T_{no} denotes non-operational time including mostly waiting times for each process.

Equation 1 is considered only for one batch scheduling problem. For actual factory data, with its inherent variations in parameter values, equation 1 can be transformed to the multiplication process:

$$MLT = n_m (T_{su} + QT_o + T_{no}) \quad (2)$$

where Q and n_m are represented by straight arithmetic averages and variables T_{su}, T_o, T_{no} are calculated as weighted-average values.

Then, the formula for calculation of average MLT, can be expressed as

$$MLT = \frac{\sum_{j=1}^{n_Q} n_{mj}}{n_Q} \left\{ \frac{\sum_{j=1}^{n_Q} n_{mj} \cdot \overline{T_{suj}}}{\sum_{j=1}^{n_Q} n_{mj}} + \frac{\sum_{j=1}^{n_Q} Q_j}{n_Q} \left(\frac{\sum_{j=1}^{n_Q} n_{mj} \cdot Q_j \overline{T_{oj}}}{n_m Q n_Q} \right) + \frac{\sum_{j=1}^{n_Q} n_{mj} \cdot \overline{T_{noj}}}{\sum_{j=1}^{n_Q} n_{mj}} \right\} \quad (3)$$

where n_Q equals the number of batches, Q_j represents the batch quantity of batch j among n_Q batches, nmj indicates the number operations in the process routing for batch j. In the weighted-average expressions individual symbols mean: $Tsuj$ - the average setup time for batch j, $Tnoj$ - average non-operation time for batch j and Toj - average operation time for batch j.

Equation 3 is usable in a case when elements of non operation times are predicable. In case when applying a parallel batch processing approach, then previous equation needs to be modified. A parallel batch scheduling assume that batches are processed on machines in smaller lots, while for a serial batch processing is typical that all components are completed at a workstation before they move to the next one. If the batches are divided into N equal-size sub-batches, the idle time becomes (Kodeekha and Somlo, 2008). Accordingly, for a given problem we divided non-operational time to two groups: the *down time waiting for parts – Tnop* and *waiting time of parts in queue - Tnoq*. Differences between these two methods are shown in figure 1.

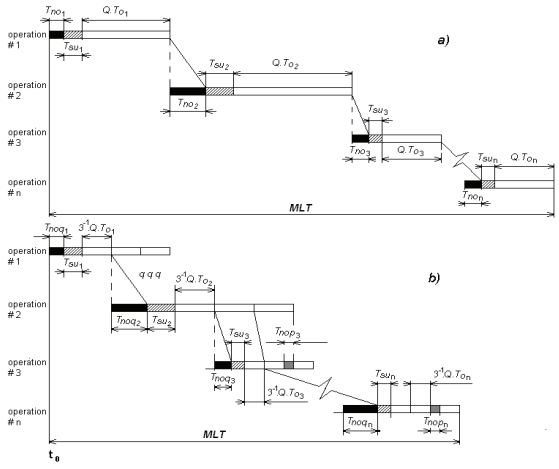


Figure 1: Time components of serial batch scheduling (a), parallel batch scheduling (b).

As is shown in figure 1b, item's manufacturing lead time of parallel batch processing legitimately consists of four components: setup time, processing time for given units in the batch, queuing time

resulting from limited capacity, and down time resulting from component unavailability. Equating an item's MLT to its average manufacturing lead time may not be the best alternative because such lead times ignore the impact of lead-time variability (Mohan and Ritzman, 1998). Following the previous assumptions, then MLT for individual batches that are processed copying approach in figure 1b, can be computed by the formula:

$$MLT_{IB} = \\ = \sum_{i=1}^{n-1}(T_{sui} + \frac{Q}{N}T_{oi} + T_{nopi} + T_{noqi}) + \mathrm{T}_{\mathrm{sun}} + QT_{on} + T_{non} + T_{nopi} \quad (4)$$

where n represents the number of operations (or machines) of individual batches and N indicates number of sub-batches obtained from batch fragmentation.

4 TESTING OF RELATIONS

Testing of relations betweens between batch sizes, batch sequencing and lead times was conducted through computer simulation using SIMPLE++ (SiMulation in Production, Logistics and Engineering) software. A simulation model was developed to calculate individual lead times under different batch sizes and batch sequencing. Simulation model was specifically created for testing real manufacturing environment in a company producing electric and pneumatic actuators. In a manufacturing company, where testing was applied, 90 different products were taken under consideration. Those products are processed on modern machine tools and another machines and equipment in a batch manner with applying sequencing based on prioritization schemes. Batches during our experiment varied from 60 to 250 parts.

Simulation model composition, respecting the main optimization criterion to minimize individual manufacturing lead times, started with definition of two groups of objects required for material flow modeling. Defined were sets of 90 parts and 68 machines with single processing and multi processing ability. Subsequently, the general and detailed model of production flows at disposal to each product was designed.

Thereafter, loading of actual time values for each product in table forms with optional attributes was performed. Subsequent defined optional attribute of parts was size of batch.

From the predefined methods, as examples, the following can be mentioned:

- data input method, by which values of times related to individual part are assigned to the pertinent machine.
- output method that is functional for the purposes to allocate part routings to machine cells in compliance with a operation sequence prescription.

Mentioned relations through simulation experiments in the following order were tested. Firstly, it was detected, how a change from a serial batch processing to a parallel batch processing can influence the manufacturing lead time duration. To test it, all batches were gradually divided into N equal-size sub-batches where for batch #1 is N=1, batch #2 is N=2, batch #3 is N=3, batch #4 is N=4, batch #5 is N=5 and for batch #6 is N=6. In this experiment whole manufacturing lead time of all batches (MLT_W) was indicated. For the calculation of MLT_W it was applied equation 4 that is sufficient to cover the whole manufacturing lead time under condition that waiting time of parts in queue $Tnoq$ is being calculated for all batches from t_0 (see in figure 1b) and that all machines and equipment are available for processing parts in time t_0 . Then the whole manufacturing lead time can be calculated by the following expression:

$$MLT_W = \max MLT_{IB} \quad (5)$$

Secondly, computer experiment was focused on learning influence of batch sequencing on whole manufacturing lead time. For this purpose, batches in above-mentioned six experiments were sequenced in two manners. In a first mode bathes were sequenced according to planned schedule. In the second mode bathes were allocated to processing machinery and equipment in a random manner.

The results from these two experiments are presented simultaneously in figure 2.

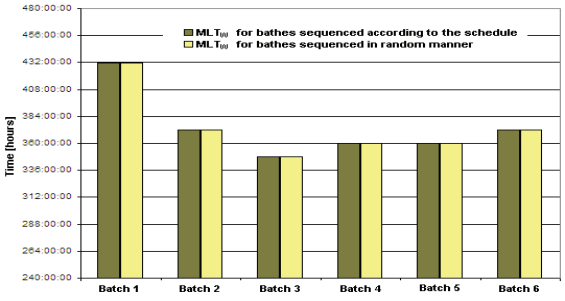


Figure 2: Whole lead times for different batches.

Another experiment was focused on comparison of MLT of selected individual batches due to the fact that changes in MLT_w between the manners of batch sequencing in the second experiment were not exposed. Therefore, individual first 5 parts for the next experiment were selected. Afterwards, batches of selected parts were gradually divided into batch #1 with N=1, batch #2 wit N=2, batch #3 with N=3 and batch #4 with N=4. Individual manufacturing lead times for given batches were calculated according to the equation 4. Evenly, as in second experiment, batches were sequenced in the same two manners. The results of these two experiments are shown in figures 3 and 4.

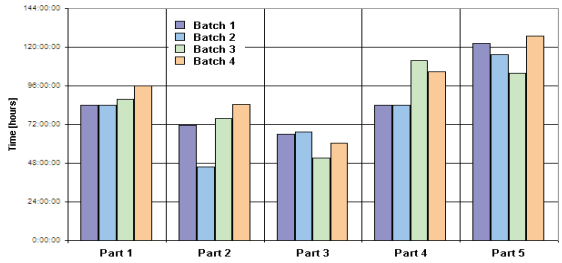


Figure 3: Individual lead times for different batches sequenced in random manner.

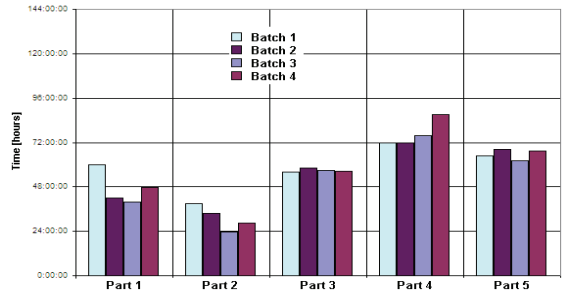


Figure 4: Individual lead times for different batches sequenced according to the schedule.

5 CLOSING REMARKS

Obtained results presented in figure 2 showed that size of batches in performed experiments influenced whole lead times. Moreover, local optimum solution of the problem between batch 2 and batch 4 can be identified. However, differences in MLT_w between batch sequencing manners in the second experiment practically were not ascertained. From the next two experiments it is possible to articulate that changes in MLT that was calculated for individual batches were influenced by different sequencing manners. Experimental results, which are demonstrated in figures 3 and 4, also showed that size of batches is influencing individual manufacturing lead times. Accordingly, in a given case there is no sense to modify sequences of batches, vice versa, it is reasonable to transform batches to the optimal sizes.

REFERENCES

Mukhopadhyay, S. K., 2008. Remanufacturing in Decentralized Supply Chains with Uncertain Yield: In *Proceedings Decision Sciences Institute 2008 Annual Meeting*, Baltimore, pp. 3531 -3536.

Akkerman, R. and van Donk, D.P., 2007. Product prioritization in a two-stage food production system with intermediate storage, *International. Journal of Production Economics*, 108, (1-2), pp. 43-53.

Fahimnia, B., Marian, R. and Motevallian, B., 2009. Analysing the hindrances to the reduction of manufacturing lead-time and their associated environmental pollution, *International Journal of Environmental Technology and Management*, Inderscience Publishers, 10, (1), pp. 16-25.

Wacker, J. G., 1996. A theoretical model of manufacturing lead times and their relationship to a manufacturing goal hierarchy, *Decision Sciences*, 27, (3), pp. 483-517.

Guiffrida, A. L. and Nagi, R., 2006. Cost characterizations of supply chain delivery performance. *International Journal of Production Economics*. 102, (1), pp.22-36.

Kuik, R. and Tielemans, P.F.J., 1999. Lead-time variability in a homogeneous queueing model of batching. *International Journal of Production Economics*, 59, (1-3), pp. 435-441.

Millar, H.H. and Yang, T., 1996. Batch sizes and lead time performance in flexible manufacturing systems. *International Journal of Flexible Manufacturing Systems*, 8, (1), pp. 5-21.

Kilpatrick, A. M., 1997. Lean manufacturing principles: a comprehensive framework for improving production efficiency. *Mechanical Engineering*. Massachusetts Institute of Technology, Cambridge, MA, February,

Groover, M. 1987., *Automation, Production & Computer Integrated manufacturing*. Prentice-Hall, USA.

Mohan R.P. and Ritzman, L.P., 1898. Planned lead times in multistage systems. *Decision Sciences* 29, (1), pp. 163–191.

Kodeekha, E. and Somlo, J., 2008. Optimal Lot Streaming for FMS Scheduling of Flow-Shop Systems, In *INES 2008 International Conference on Intelligent Engineering Systems*, Miami, Florida, pp. 53 – 58.

MULTIOBJECTIVE GA-FUZZY LOGIC CONTROLLER
Applied to a pH Reactor

Orlando Reyes
Universidad Simón Bolivar, FEI, Dpto. De Tecnología Industrial, Calle Unibolivar, Baruta, Caracas, Venezuela
orlandoreyes@usb.ve

Gustavo Sánchez, Miguel Strefezza
Universidad Simón Bolivar, MYS, Dpto. De Procesos y Sistemas, Calle Unibolivar, Baruta, Caracas, Venezuela
gsanchez@usb.ve, strefeza@usb.ve

Keywords: Genetic Algorithm, Multi-objective Optimization, Fuzzy Logic Controller.

Abstract: A Takagi-Sugeno (T-S) Fuzzy Logic Controller (FLC) is tuned using the algorithm NSGA-II. The proposed method eliminates laborious design steps such as tuning of membership functions and conclusion table parameters. An object approach representation is used to build an adequate FLC representation. Object is an individual abstraction in order to improve crossover a mutation operators. The Genetic Algorithm optimization is carry out over signal response performance parameters, in this work: settling time, rise time, overshoot and steady state error. Experiments show how the algorithm reached good response of some individuals in solution set, typically called Pareto frontier.

1 INTRODUCTION

pH control is a difficult benchmark problem due to the nonlinearity and sensibility near the neutral point. Control engineers would like to keep a desirable set-point, rejecting disturbances and tracking a reference signal.

FLC tuned via Genetic Algorithm (GA) to control a pH reactor have shown good results in the unconstrained case. In (Reyes *et al*, 2008) an object approach was proposed to obtain the FLC parameters, optimizing a scalar objective function based on the loop error. The authors have also used the sum of Rise time (Rt), Settling time (St), Overshoot (Os) and Steady state error (Sse) as a fitness function.

The indicators Rt, St, Os and Sse measure the performance of a system response and could be in possible conflict. If we try to minimize one, another or the rest of metrics could increase. Real problems involve more than one objective. Multiobjective evolutionary techniques try to find the Pareto frontier in the objective space (Coello, 2004), and the control designer has to choose the best trade-off for a given application.

2 MOEA

A multiobjective problem seeks to optimize the components of a vector-valued objective function. Unlike the single objective optimization the solution to this problem is not a single point, but a family of points know as the Pareto-optimal set (Tamaki *et al*, 1996). A multiobjective problem state can be stated as:

$$\begin{gathered} \text{Min } \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}),\ldots\ldots,f_i(\mathbf{x}),\ldots\ldots,f_n(\mathbf{x})\} \\ \text{s.t. a } \mathbf{x} \in \boldsymbol{D} \\ \boldsymbol{D} = \{\, \mathbf{x} \in R^n : g_j(\mathbf{x}) \le 0,\ j = 1,\ldots\ldots,\ J; \\ h_k(\mathbf{x}) = 0,\ k = 1,\ldots\ldots,\ K \,\} \end{gathered} \tag{1}$$

Several MOEAs have been proposed to solve problem (1). In this work we propose to use the Non dominated Search Genetic Algorithm (NSGA-II). It initiates with a random population in the search space, process follow assigning a particular rank to the sequential Pareto surfaces generated plotting $f_i(\mathbf{x})$ in the objectives space. After ranking assigned to every individual other important parameter called crowding distance tells how population density is or individuals are scattered in objective space in a particular Pareto frontier. The tournament process chooses the best individuals and after a default

number of generations the algorithm stops. NSGAII gives a set of solutions or Pareto surface. Researchers use it if objectives to be optimized are in conflict, thus no best or unique solution exist like in a Simple Genetic Algorithm where a unique solution is achieved.

3 CONTROLLER

Next, a briefing about Fuzzy Logic (FL), GA and MOEA applied to this particular problem is presented.

3.1 Fuzzy Logic

The Fuzzy Logic concept was proposed in a seminal paper written in 1965 by Lofti A. Zadeh. One of the first Fuzzy Logic Controlles (FLCs) was developed by (Mandani & Assilian, 1975), attempting control the speed of a steam engine.

An important issue in FLC design is searching for adequate and if possible good parameters for both membership functions and conclusion tables. Heuristic techniques are useful to perform this task. A FLC consists of a rule set that, in a linguistic manner, tells how the system must work. The output of the FLC will be the control action. Linguistic rules are constructed like statements, with cause and consequences, as follows:

IF cause_1 ***AND*** cause_2 ***THEN***
consequence_1 ***AND*** consequence_2

In this work the defuzzification process is carried out following the Takagi-Sugeno (T-S) method of order cero, with five membership functions for each input, and twenty five rules. Because is easy to program and is faster than Mamdani method. The FLC output is calculated using the weighted averaging defuzzification method (see eq. 2).

$$CF_i = \frac{\sum \alpha_j * C_{i,j}}{\sum \alpha_j} \qquad (2)$$

where:

$C_{i,j}$: conclusion i , rule j; α_j: activation degree of rule j;

CF_i: defuzzificated (crisp) value.

An extended FLC review, presented in (Gang Feng, 2006), gives the reader a clue of their broad application.

Rules are created with all possible combinations between Error and Derror fuzzy values. **NB**: Negative Big, **NS**: Negative Small, **Z**: Zero, **PS**: Positive Small, **PB**: Positive Big.

IF Error is ***NB AND*** Derror is **PB** ***THEN*** Ci,1
: : : : : : : : : : :
IF Error is ***PB AND*** Derror is **NB** ***THEN*** Ci,25

Next table show how to construct the rules.

Table 1: Rule table, i = 1 affects valve of acid, i = 2 otherwise.

Error / Derror	NB	NS	Z	PS	PB
PB	$C_{i,1}$	$C_{i,2}$	$C_{i,3}$	$C_{i,4}$	$C_{i,5}$
PS	$C_{i,6}$	$C_{i,7}$	$C_{i,8}$	$C_{i,9}$	$C_{i,10}$
Z	$C_{i,11}$	$C_{i,12}$	$C_{i,13}$	$C_{i,14}$	$C_{i,15}$
NS	$C_{i,16}$	$C_{i,17}$	$C_{i,18}$	$C_{i,19}$	$C_{i,20}$
NB	$C_{i,21}$	$C_{i,22}$	$C_{i,23}$	$C_{i,24}$	$C_{i,25}$

Defuzfication is done calculating CF1: conclusion at flow 1 and CF2: conclusion at flow 2, with the equation 2. Where α_j is the max value between both membership degree Error and Derror at the FLC input.

3.2 Genetic Algorithm

A Genetic Algorithm (Holland, 1975), (Golberg, 1953) is an iterative stochastic optimization process based in how the nature selects the best individual to survive within a given environment. They are now accepted by both the optimization and control communities to solve problems for which classical methods (i.e mathematical programming) can not be used or are not efficient enough. A GA starts with a scattered random population in a bounded space. An adaptation (or fitness) value is assigned to every individual. Fitness will be used to give a selection probability for crossover, survivor or mutation operations. The choice of the best individuals for crossover will give good "chromosomes" to children. Mutation prevents premature convergence relocating individuals. The process is iterated with the hope to obtain better individuals when algorithm stops.

3.3 NSGAII Procedure

Multiobjective problem (see eq. 1) start determining searching space and spreading a randomly population in it. Every individual is a FLC who's chromosomes are defined with membership functions and conclusion tables parameters. In simulation, individuals have it fitness vector composed of {Rt, St, Os, Sse} (eq. 3), those are objective space dimensions in where the individual "adaptation" is plotted.

$$\begin{aligned} \text{Min } \mathbf{f}(\mathbf{x}) &= \{\text{Rt}(\mathbf{x}), \text{St}(\mathbf{x}), \text{Os}(\mathbf{x}), \text{Sse}(\mathbf{x})\} \\ &\text{s.t. a } \mathbf{x} \in \boldsymbol{D} \\ \boldsymbol{D} &= \{\, \mathbf{x} \in \mathrm{R}^{\mathrm{n}}\} \end{aligned} \tag{3}$$

Where x are membership functions and conclusion table parameters.

Individuals of population in objective space have its particular Rank (R) and crowding distance (*cd*).

Rank is equal to one if individual belongs to Pareto Frontier (PF), later those are removed and the sequential individuals continue whit rank two and so on, this process discriminate several local PF. Rank assignment is done by PF definition (Augusto *et al*, 2006), consider two solutions vectors **x** and **y**, **x** is contained in the PF if.

$$\begin{cases} \forall i \in 1,2,...,k : f_i(x) \le f_i(y) \\ \quad and \\ \exists j \in 1,2,...,k : f_j(x) < f_i(y) \end{cases} \tag{4}$$

In the case of (4) **x** dominates **y** in the $\mathbb{R}^k$ objective space and have Rank one.

Crowding distance is the distance between one individual and two near it in the same PF (see eq. 5).

$$cd_i = \sum_{p=1}^{p=m} \sqrt{\sum_{c=1}^{c=n} \left(X_{cp} - X_{ci}\right)^2} \tag{5}$$

Where *c* is an objective space axis and *n* are the number of the objectives; *p* is a particular point and *m* are the total points in the same Pareto Frontier; *i* is the individual.

Binary selection is carry out and tournament is done first by Rank. individuals with minor Rank are preferred, if both have equal R, *cd* is taken into account, mayor *cd* wins the tournament to preserve population diversity, two individuals are then selected by this process for crossover and mutation.

Simulated binary crossover (Deb & Agrawal, 1995) makes information interchange, and to avoid premature convergence polynomial mutation works well (see eq. 6).

$$c_k = p_k + \left(p_k^u - p_k^l\right)\delta_k \tag{6}$$

where *k* is the vector *k*-component, *c* is the child, *p* the parent δ an uniform random number *u* and *l* are the upper and lower bounds in the search space.

New and old population are joined and selected via tournament to conform the new generation, and then survivors could appear. The process is repeated until reach the maximum number of iterations.

In a previous work, population of the Initial Individuals where created with restrictions in membership functions (Reyes *et al*, 2008) in hope of avoid overlapping or empty space but no restrictions where imposed while NSGAII was running, thus membership functions at the end shown empty space in discourse universe, overlapping or both mixed cases (Fig. 2,3).

4 PH REACTOR

The equations for the pH dynamic were developed in (McAvoy *et al*, 1975). The main issue is to keep the process around the neutral point, where the system is very sensitive and highly non linear, then pH control is regarded as a benchmark problem, especially when the reference signal change from pH=7 to a mayor value nearby. The interested reader can easily verify this fact by the construction of the neutralization or titration curve (TC). An experimental method to obtain the TC is based on holding the base concentration constant, slowly adding the acid and then plotting the pH versus the acid concentration. Three operating zones are commonly considered: low, medium, high (see Fig 1).

pH is usually controlled by the mixture of two solutions with different concentrations, one basic and other acid. In this work, we validated our SIMULINK® model by comparing the resulting TC with the one presented in (Zhang, 2001).

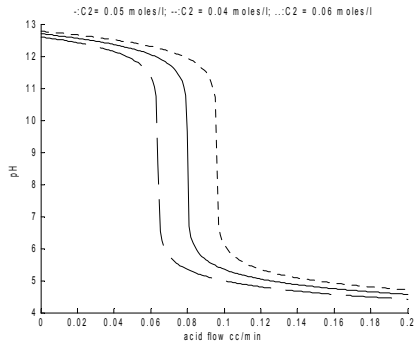


Figure 1: Titration curve, zones low, medium, high, pH approximately 0~6, 6~11.5, 11.5~14, respectively.

The neutralization process takes place within a Continuous Stirred Tank Reactor (CSTR). There are two flows to the CSTR. One is acetic acid of concentration C_1 at flow rate F_1, and the other is sodium hydroxide of concentration C_2 at flow rate F_2.

The mathematical equations of the CSTR are shown in eq's 7-12.

Table 2 shows the parameters and model variables.

Table 2: Description and values for parameters and variables.

Name	Description	Value
V	Volume of tank	1 L
F_1	Flow rate of acid	0.081 L/min
F_2	Flow rate of base	0.512 L/min
C_1	Concentration of acid in F1	0.32 mol/L
C_2	Concentration of acid in F2	0.05005 mol/L
K_a	Acid equilibrium constant	1.8×10^{-5}
K_w	Water equilibrium constant	1.0×10^{-14}
$[H^+]$	Hydrogen ion	-
[HAC]	Acetic acid	-
$[AC^-]$	Acetate ion	-
$[NA^+]$	Sodium ion	-

$$V\frac{d\xi}{dt} = F_1C_1 - (F_1 + F_2)\xi \tag{7}$$

$$V\frac{d\varsigma}{dt} = F_2C_2 - (F_1 + F_2)\varsigma \tag{8}$$

$$\begin{aligned}&\left[H^+\right]^3 + (K_a + \varsigma)\left[H^+\right]^2 + \\ &\{K_a(\varsigma - \xi) - K_w\}\left[H^+\right] - K_wK_a = 0\end{aligned} \tag{9}$$

$$pH = \log_{10}[H^+] \tag{10}$$

$$\xi = [HAC] + [AC^-] \tag{11}$$

$$\varsigma = [NA^+] \tag{12}$$

5 EXPERIMENT RESULTS

The experiment was done using MATLAB® and SIMULINK®, starting parameters are shown in table 3, they where defined by trial and error of several experiments.

Following results where obtained from PF set for the individual with minimum Sse (see Table 4). After algorithm run designer choose what is more convenient as desire system response, other values could be refined if minimum of more than one objective if required.

Table 3: Input parameters to the NSGAII.

Parameter	Value
Generations	25
Individuals	30
Crossover probability	0.8
Mutation Probability	0.01
Error Domain	[-10 10]
Rate of error Domain	[-10 10]
Conclusion Domain at inlet 1	[0 7]
Conclusion Domain at inlet 2	[0 7]

Values of membership functions parameters for error

-10.00 -5.95	Left trapeze
-9.75 -6.39 2.42	Triangle
-6.31 7.65 7.80	“
-4.79 -3.25 4.20	“
5.84 10.00	Right trapeze

Figures 2 show the final distribution of the error membership functions, in where overlapping appear in absent of restrictions while NSGAII was running.

Figures 3 show the final rate of error distribution membership functions, in where empty space and overlapping appear in absent of restrictions while NSGAII was running.

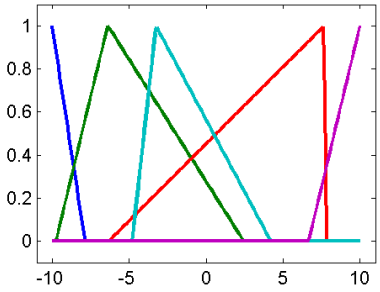


Figure 2: Error membership functions.

Values of membership functions parameters for rate of error

-7.00 -5.10	Left trapeze
-6.87 -5.86 -3.21	Triangle
0.67 0.92 2.77	“
-0.15 5.14 6.73	“
5.80 7.00	Right trapeze

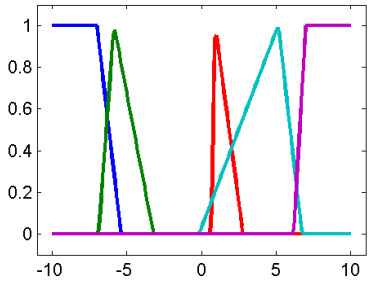


Figure 3: Rate of error membership functions.

Consequence values (according to Table 1) are used in the FLC to get and adequate system response.

Consequence matrix, flow 1				
6.23	6.98	6.17	5.17	4.39
5.35	4.93	2.46	4.25	2.48
0.97	1.04	2.95	3.61	0.18
3.01	6.20	6.22	5.31	2.72
0.48	2.73	0.84	1.14	6.48

Consequence matrix, flow 2				
5.40	0.08	2.01	5.63	5.94
1.98	0.11	3.63	5.45	5.14
5.82	5.97	6.89	2.90	6.59
6.23	0.05	0.70	2.40	0.15
6.31	3.81	4.43	0.44	4.80

Table 4: Objectives to the individual with minimum Sse NSGAII.

Objective	Value
Settling time	2.0511
Undershoot	2.5918
Overshoot	0.1158
Rise Time	3.4483e-004
Steady State Error	0.0217

System response with FLC parameters at the final of the NSGAII run do its job following reference signal and disturbance rejection near neutral point (Fig. 4-6).

Os and Sse where the only objectives in conflict as show in Fig 7. against other objectives combined and that's why there aren't shown here.

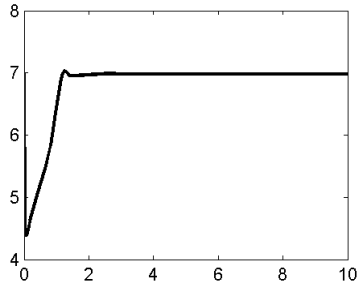


Figure 4: System response pH vs time.

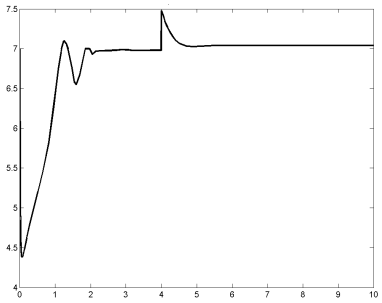


Figure 5: Disturbance Rejection.

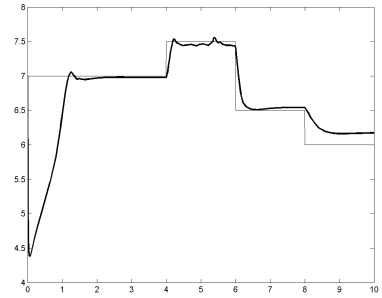


Figure 6: Signal tracking.

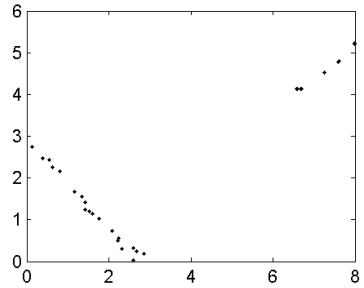


Figure 7: Space Os vs Sse.

6 CONCLUSIONS

Multiobjective Evolutionary Algorithms in the case of NSGAII is an excellent tool to find FLC membership functions and conclusion table parameters, especially is control designer wants to refine one objective on case depending. The method gives a population of FLC to choose a desired one, additionally shows what objectives are opposed.

Parameters in table 3 are subject of discussion with more intensive run of the whole algorithm.

The algorithm is very slow, every individual of population must be simulated also fitness assigned, and is necessary a lot of computer resource but is achieved of line.

7 FUTURE WORK

Basic restrictions would be imposed to obtain more homogeneous membership functions distribution, avoiding overlapping and empty space in discourse universe in membership functions.

Program Mamdani method applying restrictions in consequence and compare difference in response.

Algorithm stop criteria must be implemented to compare it with other MOEA in order to establish performance metrics.

REFERENCES

Reyes, O., Sanchez, G., Strefezza, Miguel, 2008. Usign Genetic Algorithms to design a Fuzzy Logic Controller for a pH Reactor: an Object Approach. *"IASTED International conference, Quebec City, Canada Control and Applications (CA 2008) May 26-28", Proceedings of the xxx conference IASTED*. http://www.actapress.com/Abstract.aspx?paperId=33609

L. A. Zadeh. Fuzzy Sets. 1965. *Information and Control Vol 8*. 338-353.

E. H. Mamdani and Assilian. 1975. An Experiment In Linguistic Synthesis with a Fuzzy Logic Controller. *Int. Man Mach Stud Vol. 7*. 1-13.

Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. INSTICC Press.

Smith, J., 1998. *The book*, The publishing company. London, 2nd edition.

E. H. Mandani and S. Assilian. 1975. An Experiment In Linguistic Synthesis with a Fuzzy Logic Controller, *Int. J. Man Mach. Stud. Vol. 7*. 1-13.

Holland, J. Adaptation in Natural and Artificial Systems. 1975. *University of Michigan Press, Ann Arbor.*

David E. Golberg, Genetic Algorithms in Search Optimization & Machine Learning. 1953.*Addison Wesley Longman Inc.*

Gang Feng. 2006. A Survey on Analysis and design of Model-Based Fuzzy Control Systems. *IEEE Transactions on Fuzzy Systems,14(5).*

McAvoy, T. J.; Hsu, E.; Lowenthal, S. 1975. Dynamics of pH in a Controled Stirred Tank Reactor. *Ind. Eng. Chem. Process Des. Dev. 11.* 68 –70.

Zhang, J. 2001.A Nonlinear Gain Scheduling Control Strategy Based on Neuro-Fuzzy Networks. *Ind. Eng. Chem. Res. 40.* 3164 –3170.

Carlos A. Coello Coello. 2004. Recent Trends in Evolutionary Multiobjective Optimization. *Evolutionary Multiobjective Optimization Theoretical Advances and Applications*. Springer. ISBN 1-85233-787-7.

Deb, K.; Agrawal, S.; Pratab, A. & Meyarivan, T. 2000. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multiobjective Optimization: NSGAII. *Proceedings of Parallel Problem Solving From Nature VI Conference*, pp 849-858.

O.B. Augusto, S. Rabeau, Ph. De'pince, F. Bennis. 2006. Multi-objective genetic algorithms: a way to improve the convergence rate. *Engineering Applications of Artificial Intelligence 19*, 501–510

H. Tamaki, H. Kita, and S. Kobayashi. 1996. Multi-objective optimization by Genetic Algorithms: A review. *IEEE Proc. Int. Conf. Evolutionary Computation*, 517–522

A GENERAL MODEL FOR JOB SHOP PROBLEMS USING IMUNE-GENETIC ALGORITHM AND MULTIOBJECTIVE OPTIMIZATION TECHNIQUES

Q. Zhang, H. Manier and M.-A. Manier
University of Technology of Belfort-Montbéliard, laboratory Systems and Transport
90010 Belfort Cedex, France
Tel: +33(0)3 84 58 38 35
qiao.zhang@utbm.fr, herve.manier@utbm.fr, marie-ange.manier@utbm.fr

Keywords: Shop scheduling problem with transportation constraints, Immune-genetic algorithm, Multiobjective optimization, Pareto.

Abstract: We define a global model to simulate the characteristics of three kinds of the manufacturing systems with transport resources. Based on this model, we use an immune-based genetic algorithm to solve the associated scheduling problems. We take the makespan and minimum storage as the two objectives and use modified Pareto ranking method to solve this problem. We show how to choose the best solutions for the studied systems. Though not all the constraints of the real systems are considered until now, the computational results show that our proposed model and algorithm have efficiencies in solving scheduling problems.

1 INTRODUCTION

During the past decade, problems in production planning have been arisen dramatically in automated manufacturing systems. A well planed synchronization between the machines and the transportation resources are crucial to improve their efficiency. Most of the classical works do not consider transport operations constraints. However, material handling systems may become critical resources. Moreover numerous practical constraints have to be taken into account, and several objectives have to be considered. Multiobjective optimization no doubt plays a very important role to get a more realistic solution for the decision maker.

In this paper, we consider the manufacturing systems with transportation resources which can be classified into mainly three main classes: flexible manufacturing systems (FMS), robotic cells (RC), and treatment surface facilities (TSF). A classification can be found in the literature for each of these systems (Hall et al., 1998, Tacquard & Martineau, 2001, Manier & Bloch, 2003, Brauner et al., 2005). In each system, the associated scheduling problems can be considered as specific ones. Nevertheless, there also exist similarities among them. In fact, there is few works related to the general problems which link the scheduling of product operations and transportation together. Knust (Knust, 1999) integrated the transportation issues into classical scheduling models. In the same way, we try to define a global model suitable for any of those systems with transportation constraints (section 2). An improved immune-based genetic algorithm and a modified Pareto-compliant ranking method are applied as the main solving methods to solve the scheduling problem with two objectives (minimization of the makespan and the storage) (section 3). The computational results for the proposed algorithm show that our model and the adopted algorithm are efficient enough to schedule the activities of production (section 4).

2 GENERAL MODEL

2.1 Notation

Our notations consider the following four aspects:

1) Job/task

n : total number of jobs.
O_i: number of the operations of job i ($i \in [1, n]$).
$P(i,j)$: operation j of job i ($j \in [1, O_i]$).
p_{ijk}: processing time for $P(i,j)$), on machine MP_k.
$\bar{p}_{ijk}$: minimal processing time of $P(i,j)$ on machine

MP_k.

p^+_{ijk}: maximal processing time of $P(i,j)$ on machine MP_k.

d_i: due date of job i, $i \in [1, n]$.

t_{ij}: starting date of $P(i,j)$. ($i \in [1, n], j \in [1, O_i]$)

C_i: completion time of job i.

2) Processing Resources

MP: total set of the machines (processing resources)

MP_k: machine k ($k \in [1, |MP|]$) (unitary capacity).

PR_{ij}: is the total set of the processing resources that can perform operation j of job i . ($i \in [1, n], j \in [1, O_i]$).

PJ_{ijk}: PJ_{ijk} =1, if the operation j of job i is performed by machine MP_k; PJ_{ijk} =0, otherwise

$YP_{iji'j'k}$: =1, if $P(i, j)$ is performed right before $P(i', j')$ on the machine MP_k; =0, otherwise

$S_{iji'j'k}$: setup time on MP_k between $P(i,j)$ and $P(i',j')$.

3) Transportation Resources

MT: total set of the transportations.

MT_h: transportation resource h (Unitary capacity).

$T(i, j)$: transportation task between P(i, j) and P(i, j+1).

TR_{ij}: total set of the transportation resources that can transport T(i, j).

l^-_{kh} , l^+_{kh} : needed time for a transportation resource MT_h to unload (respectively to load) machine MP_k.

$\sigma_{kk'h}$: empty travel time between machine MP_k and $MP_{k'}$ by transportation resource MT_h, k, k' $\in$ [1, |MP|], $h \in$[1, |MT|].

$\tau_{kk'h}$: loaded travel time between machines MP_k and $MP_{k'}$ by transportation resource MT_h.(it includes l^-_{kh} and l^+_{kh})

TJ_{ijh} : =1, if $T(i, j)$ is performed by MT_h; =0, otherwise.

$YT_{iji'j'h}$: =1, if MT_h performs $T(i, j)$ right before $T(i', j')$; $YT_{iji'j'h}$ =0, otherwise.

4) Storage Configuration:

γ^{s-}_{ijk} :time of the input buffer for $P(i,j)$ treated on MP_k.

γ^{s+}_{ijk} : time of the output buffer for $P(i,j)$ on MP_k.

2.2 Mathematical Model

The objectives of the general model are to minimize the makespan and the minimal storage:

$$Min \quad C\max = \underset{i=1 \ to \ n}{Max}(C_i), C_i = t_{iO_i} + \sum_{k \in PR_{ij}} p_{ijk} \times PJ_{ijk}$$

$$Min \quad \sum_{k \in MP} \sum_{i \in n} \sum_{j \in O_i} (PJ_{ijk} \times \gamma^{s-}_{ijk} + PJ_{ijk} \times \gamma^{s+}_{ijk})$$

And the following constraints of the problem are respected. $\forall i \in [1, n], \forall j \in [1, O_i - 1]$,

$$t_{ij} + \sum_{k \in PR_{ij}} p_{ijk} \times PJ_{ijk} \le t_{ij} \tag{1}$$

$\forall i \in [1, n], \forall j \in [1, O_i], \forall k \in PR_{ij}$,

$$p^-_{ijk} \le p_{ijk} \le P^+_{ijk} \tag{2}$$

$$\forall i \in [1, n], \ \forall j \in [1, O_i], \ \sum_{k \in PR_{ij}} PJ_{ijk} = 1 \tag{3}$$

$$\forall i \in [1, n], \ \forall j \in [1, O_i - 1], \ \sum_{h \in TR_{ij}} TJ_{ijh} = 1 \tag{4}$$

$\forall (i, i') \in [1, n]^2, \forall j \in [1, O_i], \forall j' \in [1, O_{i'}]$,

$\forall k \in PR_{ij} \cap PR_{i'j'}$, and M is a very large fixed number.

$$t_{ij} + p^-_{ijk} + l^-_{kh_2} + l^+_{kh_3} + s_{iji'j'k} \le t_{i'j'} + (1 - YP_{iji'j'k}) \times M \tag{5}$$

$$PJ_{ijk} \times PJ_{i'j'k} \left(t_{i'j'} + p_{i'j'k} + l^-_{kh_3} + l^+_{kh_4} + s_{i'j'ijk} \right) \le t_{ij} + YP_{iji'j'k} \times M \tag{6}$$

$\forall (i, i') \in [1, n]^2$, $\forall j \in [1, O_i]$, $\forall j' \in [1, O_{i'}]$, $\forall h \in TR_{ij} \cap TR_{i'j'}$, $PJ_{ijk} = 1$, $PJ_{i(j+1)k'} = 1$, $PJ_{i'j'k''} = 1$, $PJ_{i'(j'+1)k'''} = 1$,

$$t_{ij} + p_{ijk} + \gamma^{s+}_{ijk} + \tau_{kk'h} + \sigma_{k'k''h} \le t_{i'j'} + p_{i'j'k''} + \gamma^{s+}_{i'j'k} + (1 - YT_{iji'j'h}) \times M \tag{7}$$

$$(t_{i'j'} + p_{i'j'k''} + \gamma^{s+}_{i'j'k} + \tau_{k''k'''h} + \sigma_{k'''kh}) \times TJI_{ijh} \times TJ_{i'j'h} \le t_{ij} + p_{ijk} + \gamma^{s+}_{ijk} + YT_{i'j'ijh} \times M \tag{8}$$

$\forall i \in [1, n], \ \forall j \in [1, O_i - 1] / TJ_{ijh} = 1, \ PJ_{ijk} = 1, \ PJ_{i(j+1)k'} = 1$,

$$t_{ij} + \sum_{k \in PR_{ij}} p_{ijk} \times PJ_{ijk} + \gamma^{s+}_{ijk} + \sum_{k \in PR_{ij}} \sum_{k' \in PR_{ij+1}} \tau_{kk'h} \times PJ_{ijk} \times PJ_{ij+1k'} + \gamma^{m}_{ij+1k'} + \gamma^{s-}_{ij+1k'} \le t_{i,j+1} \tag{9}$$

Constraint (1) is the precedence constraints between two operations of job i; constraint (2) is the processing time constraints for (i,j) on machine MP_k; Constraint (3) makes sure that one operation can only be assigned to one machine; constraint (4) makes sure that one operation can only be assigned to one transportation resource; constraints (5) and (6) are the capacity constraints for each processing resource MP_k; constraints (7) and (8) are the capacity constraints for each transportation resource MT_h; constraint (9) is the travelling constraint, which expresses that a transportation resource MT_h must have enough time to move a job i between two successive operations.

3 RESOLUTION

We use an improved immune-based genetic algorithm as the training method to find nondominated solutions of the *n*-objective optimization problem.

In our case, we code the antibody into two parts. The first part is a permutation of the s transportation tasks ($s = \sum_{i \in n} (O_i - 1)$). The second part is a permutation of the m operation tasks ($m = \sum_{i \in n} O_i = s + n$).

3.1 Selection Operation

In the algorithm (Zhang et al., 2006), the affinity between antigen and antibody *v*, is defined by $ax_v = opt_v$, where opt_v is the fitness of antibody *v*.

The expected selection probability e_v of antibody *v* is calculated as: $e_v = ax_v/c_v$, where c_v is the density of antibody *v*. It can be seen from the above equation that the antibody with both high fitness and low density would have more chances to survive.

We define that antibody *v* and antibody *w* have the affinity when the following inequality is satisfied $f(v,w) < L$, where $f(v,w) = d(v,w) + |ax_v - ax_w|$, and $|ax_v\text{-}ax_w|$ is the Euclidean distance, $L = L_0 \times \exp(b \cdot T)$, $L_0 > 0$, $b > 0$, and $T > 0$ is the number of evolution generations. *L* is an increasing function of evolution generations. The antibody's diversity and density would be increased efficiently with the increase of the evolution generations and that the suppression would be more powerful to preserve high diversity. So the algorithm would have strong ability to control the reproducing process.

3.2 Learning Procedure

The whole learning process of the Pareto-imune-genetic algorithm can be described as follows:

Step 1. Initialization of the Population. All the gene bits of each antibody in the first generation are generated randomly within the feasible domain. In the initialization stage, we calculate the time windows $[\alpha_{ij}^-, \alpha_{ij}^+]$ and $[\beta_{ij}^-, \beta_{ij}^+]$, which are the earliest and the latest starting dates of each operation $P(i, j)$ and $T(i, j)$ respectively. Then, according to all the constraints, we narrow down the time windows for all the operations and transportation tasks. Firstly, we update the earliest starting dates forwardly. Secondly, we update latest starting dates backwardly.

Step 2. Calculation of the Time Windows. Then we allocate the tasks on each transport resource by randomly sequence. We do the same for each machine according to constraint (1).

After that, we verify all the time windows. If an individual is not eligible we generate a new one. We do this until we obtain an eligible individual. The initial individual is replaced with this new one.

Step 3. Fitness Calculation. We change the calculation of the new fitness as follows: $f_k = \exp(-(k-1))$, which makes the value of the first objective varies according to the rank. And we take the Pareto ranking method (Goldberg, 1989) to calculate the rank. In this paper, the two objective fitness values are defined as the makespan and the minimum storage.

Step 4. Evolution of the Population. The algorithm starts with the initial population that is generated randomly. The reproduction, crossover and mutation operators are used to produce the filial generation superior to their parents. Because it has improved the affinity calculation and makes the threshold value a dynamic parameter, it has strong ability to overcome the shortage of the tendency towards local optimum value and premature. We take the single point for crossover and the single bit for mutation. The reproduction operator is based on not only the fitness but also the density which plays an important role in diversity maintenance in immune system. The aforementioned steps are performed repeatedly until all the training data are trained completely.

4 RESULTS

Here, we take a simple example of five jobs (*n*=5), with: $O_1 = 4$, $O_2 = 3$, $O_3 = 2$, $O_4 = 4$, $O_5 = 4$, $\forall i = [1,5]$, $d_i = 15$, $r_i = 0$, $MT = \{MT_1, MT_2, MT_3\}$, $\forall i = [1,n]$, $\forall j \in [1, O_i]$, $\forall k \in PR_{ij}$, $\forall h \in [1, |MT|]$, $\forall k' \in [1, |MP|]$, $p_{ijk}^- = 1$, $p_{ijk}^+ = 3$, $\sigma_{kk'h} = \tau_{kk'h} = 1$, and $l_{kh}^-, l_{kh}^+ = 0$. The colony size is taken from 10 to 100 respectively, the max evolvement generation is 9000, crossover probability is 0.8 and mutation probability is 0.15. Other parameters are: b=0.01, l_0=0.8. We run the program for 100 times, we got the pareto solutions sets, among which the best makespan is 9 and the minimum storage is 0.

Fig.2 shows results for a population size 100,

and a evolvement generation 1000. The Pareto solutions are the rectangular solutions; the others are the dominated ones. For manufacturing systems that required no storage (like in the TSF), the solutions correspond to makespans 12, 13, 14 or 15. For other systems that allow storage, we obtain solutions with better makespan. For this example the best makespan is 9 with storage 1.

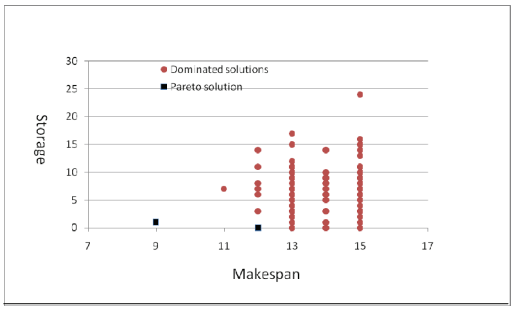


Figure 2: A resolution set with the population size 100, and with the evolvement generation 1000.

Fig. 3 and 4 respectively present a solution with and without storage. The dotted (resp. blanked) squares are transportation tasks (resp. operations); their width represents the associated times. In Fig. 3, the dotted line for P(4,2) on MP_3 means that P(4,2) can start between time 3 and time 5. The blank spaces between two transportation tasks represent the empty movements or waitness of the resource. The minimum storage corresponds to the time between T(1,3) and P(1,4) with time windows [6,7]. In Fig. 4, as all the processing times are bounded, the minimal storage for this solution can reach 0.

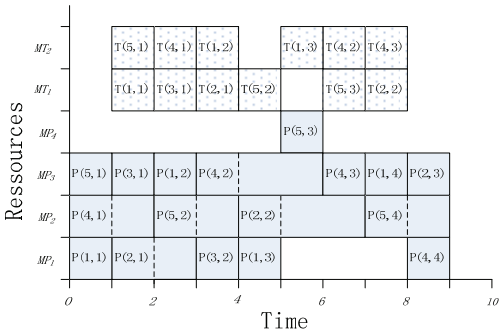


Figure 3: The time windows for a solution with makespan 9 and minimal storage 1.

5 CONCLUSIONS

We define a general model which enables us to solve several kinds of manufacturing schedule problems with transportation constraints. To reach this goal, we use pareto-immune-genetic algorithm to schedule both processing and transport operations. In this paper, we report our first results for a simplified model of a production system with or without storages, and with bounded processing times. In the future, we will complete this model with the additional constraints (the configuration of the transport network and the conflicts between transport resources). We will also try to improve our solving algorithm and to compare it with efficient algorithms developed for each of the considered systems.

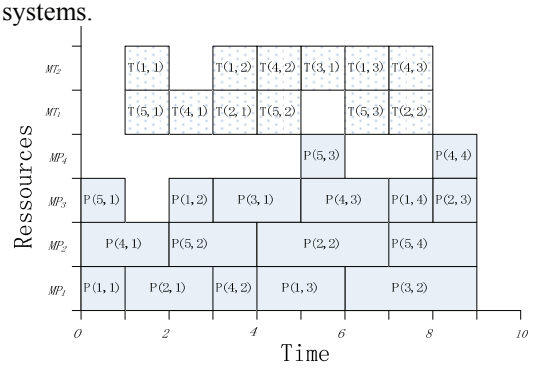


Figure 4: The time windows for a solution with makespan 9 and minimal storage 0.

REFERENCES

Brauner, N., Castagna, P., Espinouse, M., Finke, G., Lacomme, P., Martineau, P., Moukrim, A., Soukhal, A., Tacquard, C. and Tchernev, N., 2005. Ordonnancement dans les systemes flexibles de production. *RS-JESA*, 2005, 39, 925-964.

Goldberg, D. E., 1989. Genetic algorithm in search, Optimization and machine learning, *Addison-Wesley, Reading, MA*, 1989.

Hall, N. G.; Kamounb, H. & Sriskandarajah, C., 1998. Scheduling in robotic cells: Complexity and steady state analysis. *European Journal of Operational Research*, 1998, 109-1, 43-65.

Knust, S., 1999. Shop-scheduling problems with transportation. Dissertation, Fachbereich Mathematik/Informatik, Universität Osnabrück, 1999.

Manier, M.A., Bloch, C, 2003. A Classification for Hoist Scheduling Problems. *The International Journal of Flexible Manufacturing Systems*, 15(2003), 37-55.

Tacquard, C. & Martineau, P., 2001. Automatic notation of the physical structure of a flexible manufacturing system. *International journal of production economics*, 2001, 74, 279-292

Zhang, Q., Xu, X., and Liang, Y.C., 2006. Identification and speed control of ultrasonic motors based on modified immune algorithm and Elman neural networks. *Lecture Notes in Artificial Intelligence*, Vol. 4259, pp. 746-756, 2006.

WHERE WE STAND AT PROBABILISTIC REASONING

Wilhelm Rödder, Elmar Reucher and Friedhelm Kulmann
Department of Operations Research, University of Hagen, Profilstr. 8, 58084 Hagen, Germany
{*wilhelm.roedder, elmar.reucher, friedhelm.kulmann*}*@fernuni-hagen.de*

Keywords: Probabilistic Reasoning, Bayes-Nets, Entropy, MinREnt-Inference, Expert-System SPIRIT.

Abstract: Bayes-Nets are a suitable means for probabilistic inference. Such nets are very restricted concerning the communication language with the user, however. MinREnt-inference in a conditional environment is a powerful counterpart to this concept. Here conditional expressions of high complexity instead of mere potential tables in a directed acyclic graph, permit rich communication between system and user. This is true as well for knowledge acquisition as for query and response. For any such step of probabilistic reasoning, processed information is measurable in the information theoretical unit [bit]. The expert-system-shell SPIRIT is a professional tool for such inference and allows realworld (decision-)models with umpteen variables and hundreds of rules.

1 THINKING AND EXPERT SYSTEMS

1.1 From the Human Expert to his Artificial Counterpart

Humans' capabilities to memorize and recall knowledge and images, to infer facts from other facts, and to justify or explain their conclusions are admirable. The most surprising is man's ability concerning nonmonotonic reasoning: An ostrich is a bird and "all" birds fly but an ostrich does not, is contradictory but nevertheless accepted even by little childs (Rödder and Kern-Isberner, 2003a), p. 385. It was a long and a painful way for scientists to understand all such capabilities and to do first steps in the direction of modeling them. Respective studies fructified significantly artificial intelligence in its effort to simulate such phenomena on the computer. From this research resulted a great number of computer programs, called expert-systems.

1.2 Milestones in the History of Expert-Systems

After the overwhelming enthusiasm in the scientific community after the 1956 AI-workshop in Dartmouth, very famous researchers in the AI-field experimented with expert-systems like Advice Taker (1958) by McCarthy, General Problem Solver (1960s) by Newell and Simon, Mycin (1972) by Buchanan and Shortcliff, Prosepector (1979) by Duda. Further projects were Dendral, Drilling Advisor etc. For a more extensive discussion see (Harmon and King, 1985). Duda proposed a modified bayesian concept to calculate the strength of rules in a rule based system. As Duda's concept often did not show up comprehensible results a new generation of probabilistic expert-systems came up.

Scientists tried to beat the difficulties of modeling human thinking by various concepts: propositional logics, predicate logics, default logics, circumscription, conditional logics, uncertainty logics, rough sets, trues maintenance systems, among others. A still actual overview of such concepts the reader might find in (Sombe, 1990), even if already published in 1990. Only very few ideas, however, resulted in computer programs able to handle large scale knowledge domains and at the same time simulate human thinking in an adequate way.

It was in the late 1980s and in the 1990s that purely probabilistic concepts for expert-systems have been developed: HUGIN since 1989 (Hugin, 2009) and SPIRIT since 1997 (Spirit, 2009). Even if both expert systems permit probabilistic reasoning they follow absolute different philosophies, however.

2 PROBABILISTIC REASONING

2.1 Probabilistic Reasoning in Bayes-Nets

Following Jensen (Jensen, 2002), p. 19, a bayesian network is characterized as follows:

- a set of finite valued variables linked by directed edges,
- the variables and the edges form a directed acycled graph,
- to each variable with its parents there is attached a potential table,
- the variables might be of type decision variable, utility variable or state variable.

For a deeper discussion of traditional Bayes-Nets confer (Jensen, 2002). Such nets can be formed by an expert as well by empirical data. Later versions of the expert-system shell HUGIN also permit continues rather than discrete variables, only. The great advantage of such a Bayes-Net is the stringent (in)dependency-structure. This is advantageous, in as much as it forces the user to a likewise strict modeling of reality. The advantage might turn into a disadvantage when the user does not dispose of all desired probabilities. Such a model of reality feigns an epistemic state about the knowledge domain which is a biased image of reality, and consequently causes erroneous results when predicting facts from evident facts. Rödder and Kern-Isberner (Rödder and Kern-Isberner, 2003a), p.385, formulate "Inference is more. Inference is the result of the presumption and logical entailment about the vague population of our perception or even contemplation. Inference takes place in spite of incomplete information about this population." Probabilistic reasoning under maximum entropy and/or minimum relative entropy, respectively, is a promising alternative to overcome this flaw.

2.2 Probabilistic Reasoning under MinREnt

To build a knowledge base it needs a finite set of finite valued variables $\mathbf{V} = \{V_1, ..., V_L\}$ with respective values v_l of V_l. The variables might be of type type boolean, nominal or numerical. From literals of the form $V_l = v_l$, propositions $A, B, C, ...$ are formed by the junctors $\wedge$ (and), $\vee$ (or), $\neg$ (not) and by respective parentheses. Conjuncts of literals such as $\mathbf{v} = v_1, ..., v_L$ are elementary propositions, $\mathbf{V}$ is the set of all $\mathbf{v}$. $|$ is the directed conditional operator; formulas as $B|A$ are conditionals. The degree to which such conditionals are true in the knowledge domain might be expressed by probabilities $x \in [0;1]$; such conditionals or facts we write $B|A\,[x]$. As to the semantics a model is a probability distribution P for which such conditional information is valid.
More precisely, probabilistic reasoning under MinREnt is realized as follows (Rödder et al., 2006):

1. Definition of the knowledge domain
   Specification of the variables V_l and their respective values v_l, providing the set of all complete conjuncts $\mathbf{v}$.
2. Knowledge Acquisition
   Knowledge acquisition bases on a set of conditionals or facts $\mathsf{R} = \{B_i|A_i\,[x_i], i = 1, \ldots, \mathrm{I}\}$, provided by the user. The solution
   $$P^* = \arg\min R(Q, P^0), \quad s.t.\ Q \models \mathsf{R} \qquad (1)$$
   is an epistemic state among all Q which minimizes the relative entropy or Kullback-Leibler-divergence R from P^0, satisfying the restrictions R. P^* obeys the MinREnt-principle, in that it respects R without adding any unnecessary information (Rödder and Kern-Isberner, 2003b), p. 467. Bear in mind that for a uniform P^0, minimizing the relative entropy (1) is equivalent to maximizing the entropy $H(Q)$. For more details about the principles MinREnt and MaxEnt and their axiomatic foundations, the reader is referred to (Kern-Isberner, 1998), (Shore and Johnson, 1980).
3. Query
   The query process consists of three steps: focus, adaptation to the focus and question plus response. A focus specifies a temporary assumption about the domain represented by a set of conditionals or facts $\mathsf{E} = \{F_j|E_j\,[y_j], j = 1, \ldots, \mathrm{J}\}$. The adaptation of P^* to E yields the solution
   $$P^{**} = \arg\min R(Q, P^*), \quad s.t.\ Q \models \mathsf{E}. \qquad (2)$$
   Finally for a question $H|G$ under the facts R and the focus E, the answer is
   $$P^{**}(H|G) = z. \qquad (3)$$

The three-step process (1), (2), (3) is called MinREnt inference process (Rödder and Kern-Isberner, 2003b), p. 467. All values of the objective functions in the three steps −as well as the entropies $H(P^0)$, $H(P^*)$, $H(P^{**})$− measure in [bit]; the lower entropy the richer acquired knowledge about the domain. This proximaty to information theory is essential but can not be developed here. For an extensive discussion cf.(Rödder and Kern-Isberner, 2003a).

3 WHAT IS THE ADVANTAGE OF MinREnt OVER BAYES-NETS?

In this section we want to justify our position that MinREnt is better than Bayes-Nets.

- Already the propositions A,B and the conditionals $B|A$ may be pretty complex, due to arbitrary combinations of literals by the conjuncts $\wedge,\vee,\neg$. Handling such expressions in Bayes-Nets is impossible.
- Moreover, syntactical formulas like$(B|A)\wedge(D|C)$, $(B|A)\vee(D|C)$, $\neg(B|A)$, $(B|A)|(D|C)$, so called composed conditionals, allow a rich linguistic semantics on a domain, near human language. For a deeper discussion , cf. (Rödder and Kern-Isberner, 2003a), p. 387. Are already neither general propositions nor conditionals representable in Bayes-Nets the less are composed conditionals.
- The formulation of cyclic dependencies between propositions, e.g. $B|A,C|B,A|C$ is possible in MinREnt-inference. Such dependencies are not permitted in Bayes-Nets as they are DAGs.
- Bayes-Nets suffer from certain difficulties when there is a multiple functional dependence between input variables and an output variable. Such a situation forces the user to additional constructions like "noisy-and" or "noisy-or" (Diez and Galan, 2003). In a MinREnt and conditional environment such dependencies are simply and solely formulated as conditionals and the rest is done by the entropy.

All such advantages over Bayes-Nets, of course, must be accompanied by some disadvantages. Because of the absolute freedom in formulating rules, for the unexperienced user there is a high risk to cause inconsistencies: Equation (1) is not solvable. To overcome this problem, SPIRIT allows for solving the inconsistency problem in that it offers slightly modified probabilites $x_i^{'}$ instead of x_i for (1). And the user might decide if he or she accepts these probabilities or not.
Usually a set R of rules does not fully determine the epistemic state P over a domain. The freedom to admit imperfect information in R has its price. This price is a possible unreliability of the answer (3). SPIRIT informs the user about such unreliability or second order uncertainty, and invites him/her to add further information.

4 PROFESSIONALITY OF SPIRIT

SPIRIT is a professional expert-system-shell, allowing for the implementation of middle and large scaled knowledge bases. For the reader familiar with probabilistic inference models, first designed for Bayes-Nets, (Hugin, 2009), we list a few examples which where adapted to SPIRIT. Note that the stringent syntax in Bayes-Nets is overcome in SPIRIT. But vice versa, any Bayes-Net application can be modeled in the shell. The models blue baby (BB), troubleshooter (TS), and car repair (CR) are well known, (Breese and Heckerman, 1996), (Hugin, 2009). There are two models in which utility and decision variables are explicitly involved, namely the well known oil drilling problem (OD) and a credit worthiness support system (CW)(Raiffa, 1990). Besides all well known applications an outstanding knowledge base of a business-to-business approach (BS)) was modeled in SPIRIT. The latter with 86 variables and 1051 rules, partly cyclic. Knowledge acquisition for all the models counted in milliseconds (Rödder et al., 2006). All models are available at (Spirit, 2009) and can be tested by the reader. In Table 1 we provide a few data concerning these models. For models with up to umpteen variables and hundreds of rules a suitable form of user interface is necessary so as to inform about the knowledge structure and the inference process.

Table 1: Data for middle and large-scale models, implemented in SPIRIT.

Model	no. variables	no. rules	no. LEGs	$H(P^0)$ [bit]	$H(P^*)$ [bit]
BB	20	340	17	29.91	18.57
TS	76	574	50	76.00	12.83
CR	18	38	13	22.68	6.00
BS	86	1051	36	104.79	87.12
OD	6	18	3	8.17	4.08
CW	10	31	6	11.00	7.38

For this purpose the shell SPIRIT disposes of various communication tools: A list of all variables and their attributes, a list of all conditionals provided by the user, a dependency graph showing the Markov-Net of all stochastic dependencies between such variables, the junction-tree of variable clusters −so called Local Event Groups LEGs− indicating the factorization of the global by marginal distributions, among others (Rödder et al., 2006).

5 CONCLUSIONS AND THE ROAD AHEAD

Knowledge processing in a conditional and probabilistic environment under maximum entropy and

minimum relative entropy, respectively, is a powerful instrument supporting the user in various economical and technical decision situations. SPIRIT is a comfortable and professional shell for such knowledge processing.

Recent developments in the field are in test-stage, such as:

- an adaptation of the system to permit simulations of cognitive processes, (Rödder and Kulmann, 2006),
- the calculation of the transinformation or synentropy between arbitrary groups of variables.

Actual research activities are:

- the removal of an eventual unreliability of answers by the initiation of a self-learning process. The theoretical basis for this concept was published already in 2003 (Rödder and Kern-Isberner, 2003b), the implementation in SPIRIT is in the pipeline,
- handling of a mixture of continuous and discrete variables (Singer, 2008).

With such features the expert-system-shell SPIRIT will become even more user-friendly and will enable scientific work as well as applications in various disciplines.

REFERENCES

Breese, J. S. and Heckerman, D. (1996). *Decision-Theoretic Troubleshooting: A Framework for Repair and Experiment*. Morgan Kaufman, San Francisco.

Diez, F. J. and Galan, S. (2003). *Efficient computation for the noisy MAX.* International Journal of Intelligent System, 18, 165–177.

Harmon, P. and King, D. (1985). *Expert Systems.* Wiley, New York.

Hugin (2009). http://www.hugin.com.

Jensen, F. V. (2002). *Bayesian Networks and Decision Graphs.* Springer, New York.

Kern-Isberner, G. (1998). *Characterizing the principle of minimum cross-entropy within a conditional-logical framework.* Artificial Intelligence, New York, 169–208.

Raiffa, H. (1990). *Decision Analysis: Introductory, Lectures on Choice under Uncertainty*. Addison-Wesley, Reading, Massachusetts.

Rödder, W. and Kern-Isberner, G. (2003a). *From Information to Probability - An Axiomatic Approach.* International Journal of Intelligent Systems, 18-4, 383–403.

Rödder, W. and Kern-Isberner, G. (2003b). Self learning or how to make a knowledge base curious about itself. In *German Conference on Artificial Intelligence (KI 2003)*. LNAI 2821, Springer, Berlin, 464–474.

Rödder, W. and Kulmann, F. (2006). *Recall and Reasoning - an Information Theoretical Model of Cognitive Processes.* Information Sciences, 176-17, 2439–2466.

Rödder, W., Reucher, E., and Kulmann, F. (2006). *Features of the Expert-System Shell SPIRIT.* Logic Journal of the IGPL, 14-3, 483–500.

Shore, J. E. and Johnson, R. W. (1980). *Axiomatic Derivation of the Principle of Maximum Entropy and the Principle of Minimum Cross Entropy.* IEEE, Trans. Information Theory, 26-1, 26–37.

Singer, H. (2008). *Maximum Entropy Inference for Mixed Continuous-Discrete Variables.* Faculty of Business Science, University of Hagen, Discussion Paper 432.

Sombe, L. (1990). *Reasoning under Incomplete Information in Artificial Intelligence.* Wiley, New York.

Spirit (2009). http://www.xspirit.de.

AUTHOR INDEX

AUTHOR INDEX (CONT.)

Proceedings of ICINCO 2009
6th International Conference on Informatics in Control, Automation and Robotics - Volume 1
ISBN: 978-989-8111-99-9
http://www.icinco.org