

Xiao-Zhi Gao
António Gaspar-Cunha
Mario Köppen
Gerald Schaefer
Jun Wang (Eds.)

Soft Computing in Industrial Applications

Advances in Intelligent and Soft Computing

Editor-in-Chief

Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw
Poland
E-mail: kacprzyk@ibspan.waw.pl

Further volumes of this series can be found on our homepage: springer.com

Vol. 60. Z.S. Hippe,
J.L. Kulikowski (Eds.)
*Human-Computer Systems
Interaction, 2009*
ISBN 978-3-642-03201-1

Vol. 61. W. Yu, E.N. Sanchez (Eds.)
*Advances in Computational
Intelligence, 2009*
ISBN 978-3-642-03155-7

Vol. 62. B. Cao,
T.-F. Li, C.-Y. Zhang (Eds.)
*Fuzzy Information and
Engineering Volume 2, 2009*
ISBN 978-3-642-03663-7

Vol. 63. Á. Herrero, P. Gastaldo,
R. Zunino, E. Corchado (Eds.)
*Computational Intelligence in Security for
Information Systems, 2009*
ISBN 978-3-642-04090-0

Vol. 64. E. Tkacz, A. Kapczynski (Eds.)
*Internet – Technical Development and
Applications, 2009*
ISBN 978-3-642-05018-3

Vol. 65. E. Kącki, M. Rudnicki,
J. Stempczyńska (Eds.)
Computers in Medical Activity, 2009
ISBN 978-3-642-04461-8

Vol. 66. G.Q. Huang,
K.L. Mak, P.G. Maropoulos (Eds.)
*Proceedings of the 6th CIRP-Sponsored
International Conference on Digital
Enterprise Technology, 2009*
ISBN 978-3-642-10429-9

Vol. 67. V. Snášel, P.S. Szczepaniak,
A. Abraham, J. Kacprzyk (Eds.)
*Advances in Intelligent Web
Mastering - 2, 2010*
ISBN 978-3-642-10686-6

Vol. 68. V.-N. Huynh, Y. Nakamori,
J. Lawry, M. Inuiguchi (Eds.)
*Integrated Uncertainty Management and
Applications, 2010*
ISBN 978-3-642-11959-0

Vol. 69. E. Piętka and J. Kawa (Eds.)
*Information Technologies in
Biomedicine, 2010*
ISBN 978-3-642-13104-2

Vol. 70. Y. Demazeau, F. Dignum,
J.M. Corchado, J. Bajo Pérez (Eds.)
*Advances in Practical Applications of Agents
and Multiagent Systems, 2010*
ISBN 978-3-642-12383-2

Vol. 71 Y. Demazeau, F. Dignum,
J.M. Corchado, J. Bajo, R. Corchuelo,
E. Corchado, F. Fernández-Riverola,
V.J. Julián, P. Pawlewski, A. Campbell (Eds.)
*Trends in Practical Applications of Agents
and Multiagent Systems, 2010*
ISBN 978-3-642-12432-7

Vol. 72. J.C. Augusto, J.M. Corchado,
P. Novais, C. Analide (Eds.)
*Ambient Intelligence and Future
Trends, 2010*
ISBN 978-3-642-13267-4

Vol. 73. J.M. Corchado, P. Novais,
C. Analide, J. Sedano (Eds.)
*Soft Computing Models in Industrial and
Environmental Applications, 5th International
Workshop (SOCO 2010), 2010*
ISBN 978-3-642-13160-8

Vol. 74. M.P. Rocha, F.F. Riverola, H. Shatkey,
J.M. Corchado (Eds.)
Advances in Bioinformatics
ISBN 978-3-642-13213-1

Vol. 75. X.Z. Gao, A. Gaspar-Cunha,
M. Köppen, G. Schaefer, and J. Wang (Eds.)
Soft Computing in Industrial Applications
ISBN 978-3-642-11281-2

Xiao-Zhi Gao, António Gaspar-Cunha,
Mario Köppen, Gerald Schaefer,
and Jun Wang (Eds.)

Soft Computing in Industrial Applications

Algorithms, Integration, and Success Stories

Editors

Dr. Xiao-Zhi Gao
Department of Electrical Engineering
Helsinki University of Technology
Otakaari 5 A
FI-02015 TKK Espoo, Finland
E-mail: gao@cc.hut.fi

Dr. Gerald Schaefer
Department of Computer Science
Loughborough University, Loughborough
LE11 3TU, UK
E-mail: g.schaefer@lboro.ac.uk
gerald.schaefer@ieee.org

Dr. António Gaspar-Cunha
IPC/I3N - Institute of Polymers and
Composites University of Minho
Campus de Azurém
4800-058 Guimarães, Portugal
E-mail: agc@dep.uminho.pt

Dr. Jun Wang
William M.W. Mong Engineering
Building
Department of Mechanical and
Automation Engineering
Faculty of Engineering
The Chinese University of Hong Kong
Shatin, New Territories, Hong Kong
E-mail: jwang@mae.cuhk.edu.hk

Dr. Mario Köppen
Network Design and Research Center
Kyushu Institute of Technology
680-4, Kawazu, Iizuka
Fukuoka 820-8502, Japan
E-mail: mkoeppe@ieee.org

ISBN 978-3-642-11281-2

e-ISBN 978-3-642-11282-9

DOI 10.1007/978-3-642-11282-9

Advances in Intelligent and Soft Computing

ISSN 1867-5662

Library of Congress Control Number: 2010929203

© 2010 Springer-Verlag Berlin Heidelberg

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable for prosecution under the German Copyright Law.

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

Typeset & Cover Design: Scientific Publishing Services Pvt. Ltd., Chennai, India.

Printed on acid-free paper

5 4 3 2 1 0

springer.com

Preface

WSC 14 Honorary Chair's Welcome Message

It is my great pleasure to welcome you to participate in the 14th Online World Conference on Soft Computing in Industrial Applications (WSC14) to take place on the Internet. As a nonconventional conference, WSC14 provides a new type of platform for our participants to share and disseminate our new research findings. Although we could not meet or talk face to face at WSC14 as at conventional conferences, we could interact at our desktops/laptops in live sessions or discussion forums via the Internet during the conference period and beyond. WSC14 serves as a good complement of conventional conferences (such as WCCI) in the fields covering the whole spectrum of soft computing and its applications. In a sense, the online conference could be more efficient in terms of savings of traveling time and costs to get together at a venue.

I thank you for your active participation and wish you to enjoy the conference with fruitful results.

WSC14 Honorary Chair

Jun Wang

The Chinese University of Hong Kong

WSC 14 Chair's Welcome Message

Dear Colleague,

On behalf of the Organizing Committee, it is our honor to welcome you to the WSC14 (14th Online World Conference on Soft Computing in Industrial Applications).

A tradition started over a decade ago by the World Federation of Soft Computing (<http://www.softcomputing.org>) is continued to bring together researchers interested in advancing state of the art in the field. Continuous technological improvements since then continue to make this online forum a viable gathering format for a world class conference.

The program committee received a total of 62 submissions from more than 20 countries, covering all the continents. This reflects the worldwide nature of this event. Each paper was peer reviewed by referees, culminating in the acceptance of 33 papers for publication. Authors of all accepted papers were notified to prepare and submit their final manuscripts and conference presentations.

The organization of the WSC14 conference is completely voluntary. The review process required an enormous effort from the members of the International Program Committee. We would like to thank all the members of the International Program Committee for their contribution to the success of this conference. We would like to express our sincere thanks to the plenary presenter David Prokhorov for his excellent contribution on "Computational Intelligence in Automotive Applications," and to the special session organizers Muhammad Sarfraz for organizing "Special Session Soft Computing in Computer Graphics, Imaging, and Vision," Sudhir Kumar Barai for organizing "Special Session Soft Computing in Civil Engineering," Oscar Castillo and Patricia Melin for organizing "Special Session Soft Computing for Intelligent Control," and Michele Ottomanelli and Mauro Dell'Orco for organizing "Special Session Traffic and Transportation Systems." Also our thanks to the publisher Springer for their hard work and support in organizing the conference. Finally, we would like to thank all the authors for their high quality contributions. Congratulations especially to Roberto Sepulveda, Oscar Montiel-Ross, Oscar Castillo, and Patricia Melin to the WSC14 Best Paper Award contribution "Embedding a KM Type Reducer for High Speed Fuzzy Controller Into an FPGA." It is all of you who make this event possible!

We hope you will enjoy the conference, and we are looking forward to meeting you virtually at the WSC14. We encourage you to take an active part in the WSC14 paper discussions - your feedback is very important to other authors!

General Chairman

Mario Köppen

Kyushu Institute of Technology, Japan

Programme Co-Chairs

António Gaspar-Cunha

University of Minho, Guimarães, Portugal

Xiao-Zhi Gao

Helsinki University of Technology, Espoo,
Finland

8th February 2010

<http://wsc14.science-city.org>

Welcome Note by the World Federation on Soft Computing (WFSC) Chairman

On behalf of the World Federation on Soft Computing (WFSC) I would like to thank you for your contribution to WSC14! The 14th online World Conference on Soft Computing in Industrial Applications provides a unique opportunity for soft computing researchers and practitioners to publish high quality papers and discuss research issues in detail without incurring a huge cost. The conference has established itself as a truly global event on the Internet. The quality of the conference has improved over the years. The WSC14 conference has covered new trends in soft computing to state of the art applications. The conference has also added new features such as community tools, syndication, and multimedia online presentations.

I would also like to take this opportunity to thank the organisers for a successful conference. The quality of the papers is high and the conference was very selective in accepting the papers. Also many thanks to the authors, reviewers, sponsors and publishers of WSC14! I believe your hard work has made the conference a true success.

Chairman of WFCS

Professor Rajkumar Roy

World Federation on Soft Computing (WFSC)

8th February 2010

WSC14 Plenary Presentation

Computational Intelligence in Automotive Applications

David Prokhorov

Toyota Research Institute NA, Ann Arbor, Michigan

Abstract. Computational intelligence is traditionally understood as encompassing artificial neural, fuzzy and evolutionary methods and associated computational techniques. Nowadays there is no sharp boundary between CI and other learning methods. Different CI methodologies often get combined with each other and with non-CI methods to achieve superior results in various applications. In this presentation I will discuss CI methodological issues and illustrate them with several applications from the areas of vehicle manufacturing, vehicle system monitoring and control, as well as active safety. These will be representative of CI applications in the industry and beyond. I will also discuss some lessons learned about successful and yet-to-be-successful industrial applications of CI.

Dr. Danil Prokhorov began his technical career in St. Petersburg, Russia, in 1992. He was a research engineer in St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences. He became involved in automotive research in 1995 when he was a Summer intern at Ford Scientific Research Lab in Dearborn, MI. In 1997 he became a Ford Research staff member involved in application-driven research on neural networks and other machine learning methods. While at Ford, he took active part in several production-bound projects including neural network based engine misfire detection. Since 2005 he is with Toyota Technical Center, Ann Arbor, MI, overseeing important mid- and long-term research projects in computational intelligence. He has more than 100 papers in various journals and conference proceedings, as well as several inventions, to his credit. His personal home page is <http://home.comcast.net/~dvp/>

International Advisory Board

Nik Kasabov	Auckland University, New Zealand
Janusz Kascprzyk	Polish Academy of Sciences, Poland
Seppo Ovaska	Helsinki University of Technology, Finland
Andreas König	University of Kaiserslautern, Germany
Ashraf Saad	Armstrong Atlantic State University, USA
Jörn Mehnen	Cranfield University, UK
Bernadetta Kwintiana Ane	Stuttgart University, Germany

International Technical Program Committee (in alphabetic order)

Janos Abonyi	University of Pannonia, Hungary
Ajith Abraham	NTNU, Norway
Akira Asano	Hiroshima University, Japan
Erel Avineri	University of the West of England, UK
Sudhirkumar V Barai	IIT Kharagpur, India
Hélio Barbosa	LNCC, Brazil
Valeriu Beiu	United Arab Emirates University, UAE
Zvi Boger	OPTIMAL - Industrial Neural Systems Ltd., Israel
Oscar Castillo	Tijuana Institute of Technology, Mexico
Sung-Bae Cho	Yonsei University, South Korea
Leandro Coelho	Pontifical Catholic University of Parana, Brazil
Carlos Coello Coello	CINVESTAV-IPN, Mexico
Lino Costa	University of Minho, Portugal
Keshav Dahal	University of Bradford, UK
Justin Dauwels	Massachusetts Institute of Technology, USA
Guy De Tre	Ghent University, Belgium
Giuseppe Di Fatta	The University of Reading, UK
Carlos Fonseca	Universidade do Algarve, Portugal
Matjaz Gams	Jozef Stefan Institute, Slovenia
Bernard Grabot	ENIT, France
Roderich Groß	EPFL, Switzerland
Jerzy Grzymala-Busse	University of Kansas, USA
Ioannis Hatzilygeroudis	University of Patras, Greece
Francisco Herrera	University of Granada, Spain
Frank Hoffmann	TU Dortmund, Germany
Hisao Ishibuchi	Osaka Prefecture University, Japan
Akimoto Kamiya	Kushiro National College of Technology, JAPAN
Petra Kersting	TU Dortmund, Germany
Frank Klawonn	University of Applied Sciences BS - WF, Germany
Andreas König	TU Kaiserslautern/ISE, Germany

Renato Krohling	UFES, Brazil
Heitor Lopes	UTFPR, Brazil
Christophe Marsala	Universite Pierre et Marie Curie - Paris 6, France
Jörn Mehnert	Cranfield University, UK
Patricia Melin	Tijuana Institute of Technology, Mexico
Sanaz Mostaghimi	University of Karlsruhe, Germany
Jae Oh	Syracuse University, USA
Pedro Oliveira	University of Minho, Portugal
Marcin Paprzycki	IBS PAN and WSM, Poland
David Pelta	University of Granada, Spain
Petrica Pop	North University of Baia Mare, Romania
Radu-Emil Precup	Politehnica University of Timisoara, Romania
Ashraf Saad	Armstrong Atlantic State University, USA
Muhammad Sarfraz	Kuwait University, Kuwait
Abhinav Saxena	NASA Ames Research Center, USA
Giovanni Semeraro	Univ. of Bari Aldo Moro, Italy
Yos Sunitiyoso	University of Southampton, UK
Ricardo Takahashi	Universidade Federal de Minas Gerais, Brazil
Ashutosh Tiwari	Cranfield University, UK
Heike Trautmann	TU Dortmund, Germany
Eiji Uchino	Yamaguchi University, Japan
Olgierd Unold	Wroclaw University of Technology, Poland
Marley Vellasco	PUC-Rio, Brazil
Tobias Wagner	TU Dortmund, Germany
Xiaolei Wang	Helsinki University of Technology, Finland

Additional Reviewer

Anna Lisa Gentile

WSC 14 Technical Sponsors

WSC 14 was supported by:

World Federation on Soft Computing (WFSC)
 International Fuzzy Systems Association (IFSA)
 MIRLabs
 Fuzzy Logic Systems Institute (FLSI)
 International Neural Networks Society (INNS)
 Komunitas Soft Computing Indonesia
 Technical Committee Industrial Applications, IEEE Systems, Man, &
 Cybernetics Society
 German Chapter IEEE Computational Intelligence Society
 Elsevier B.V.

List of Contributors

János Abonyi

Universiy of Pannonia
Veszprem, Egyetem str. 10, H-8200
Hungary
abonyij@fmt.uni-pannon.hu

Luis T. Aguilar

Instituto Politecnico Nacional
Av. Del Parque 1310, Mesa de Otay
Tijuana, 22510, Mexico
luis.aguilar@ieee.org

Malidi Ahamadi

Department of Applied Mathematics,
University of Leeds
Leeds, LS2 9JT
United Kingdom
malidi@maths.leeds.ac.uk

Mohammad Reza Akbarzadeh**Totonchi**

Ferdowsi University
Vakil Abad, Mashhad,
Iran
akbarzadeh@ieee.org

Hameed AlQueri

Department of Quantitative Methods
and Information Systems,
Kuwait University, Kuwait

Mahdi Aliyari Shoorehdeli

K. N. Toosi University of Technology
Seyyed Khandan Bridge
Tehran, Iran
m_aliyari@eetd.kntu.ac.ir

Leonardo Araújo

CEFALA
Federal University of Minas Gerais
Belo Horizonte, MG 31270-901,
Brazil
leolca@gmail.com

Jawad Usman Arshed

Department of Computer Science,
CIIT
COMSATS, Abbottabad
Pakistan
jawadusman@ciit.net.pk

Vahid Asadpour

Sajad University
84 Emamat, Mashhad
Iran
asadpour@aut.ac.ir

Kambiz Badie

Iran Telecom Research Center
Tehran 14155-3961, Iran
k_badie@itrc.ac.ir

Sheela Rani Balasubramaniam
 Sathyabama University
 Jeppiaar Nagar, Rajiv Gandhi Road
 Chennai-600119, India
 kavi_sheela@yahoo.com

Raj Baldev
 Indira Gandhi Centre for Atomic
 Research
 Kalpakkam
 Chennai-603102, India
 dir@igcar.gov.in

Zoltán Bankó
 Department of Process Engineering,
 University of Pannonia
 Veszprem, P.O.Box. 158
 H-8200, Hungary

S.V. Barai
 IIT Kharagpur
 Kharagpur 721 302
 India
 skbarai@civil.iitkgp.ernet.in

Bernardo de Carvalho
 Optimization Department
 ABG Information Systems
 Belo Horizonte, MG 31170-000,
 Brazil
 bpenna@gmail.com

Patrick Borges
 Federal University of Espírito
 Santo - UFES
 Department of Statistics,
 Av. Fernando Ferrari 514,
 Campus de Goiabeiras
 Vitória - ES,
 CEP 29075-910, Brazil
 patrick@cce.ufes.br

Leonardo Caggiani
 Technical University of Bari
 Via Orabona, 4
 Bari 70125, Italy
 l.caggiani@poliba.it

Mauro Campos
 Federal University of Espírito
 Santo - UFES Department of
 Statistics
 Av. Fernando Ferrari 514,
 Campus de Goiabeiras
 Vitória - ES,
 CEP 29075-910, Brazil
 maurocmc@cce.ufes.br

Oscar Castillo
 Instituto Tecnológico de Tijuana
 Calzada Tecnológico s/n,
 Fracc. Tomas Aquino
 Tijuana B.C., 22414, Mexico
 ocastillo@hafsamx.org

Nohe R. Cazarez-Castro
 Universidad Autónoma
 de Baja California
 Calzada Universidad 14418,
 PI Internacional Tijuana
 Tijuana, 22414, Mexico
 nohe@ieee.org

José Covas
 IPC/I3N - Institute of Polymers
 and Composites, University of Minho
 Campus de Azurem
 Guimarães, 4800-058, Portugal
 jcovas@dep.uminho.pt

Ashraf Darwish
 Computer Science Department,
 Helwan University
 Cairo
 Egypt
 amodarwish@yahoo.com

João Duarte
 Department of Physics,
 Instituto Superior
 de Engenharia do Porto
 R.S. Tomé
 Porto, 4200, Portugal
 jmmd@isep.ipp.pt

António Gaspar-Cunha

IPC/I3N - Institute of Polymers
and Composites, University of Minho
Campus de Azurem
Guimarães, 4800-058, Portugal
agc@dep.uminho.pt

Aboul Ella Hassanien

Information Technology Department,
Cairo University, Giza
Egypt
aboitcairo@gmail.com

Takafumi Hiro

School of Medicine, Nihon University
30-1 Oyaguchi Kami-cho
Itabashi-ku 173-8610, Japan

Giuseppe Iannucci

Technical University of Bari
Via Orabona, 4
Bari 70125, Italy
g.iannucci@poliba.it

Shohei Ichiyama

Graduate School of Science
and Engineering,
Yamaguchi University
1677-1 Yoshida
Yamaguchi 753-8512, Japan
ichiyama@
ic.sci.yamaguchi-u.ac.jp

Muhammad Iqbal

Department of Computer Science,
CIIT COMSATS, Abbottabad
Pakistan
iqbal@ciit.net.pk

Kuncup Iswandy

Technische Universität
Kaiserslautern,
Integrierte Sensorsysteme
Erwin-Schrödinger-Strasse
Kaiserslautern, 67653,
Germany

Andreas König

Technische Universität
Kaiserslautern,
Integrierte Sensorsysteme
Erwin-Schrödinger-Strasse
Kaiserslautern, 67653,
Germany

Iman Kamkar

Islamic Azad University,
Department of Computer Science
Ostad Usafi Ave., Ghasemabad
Mashhad Branch,
413-91735, Iran
kamkar.iman@gmail.com

Takanori Koga

Fuzzy Logic Systems Institute
680-41 Kawazu, Iizuka
Fukuoka 820-0067, Japan
koga@flsi.or.jp

Andrew Koh

Institute for Transport Studies
University of Leeds
Leeds, West Yorkshire LS2 9JT
United Kingdom
a.koh@its.leeds.ac.uk

Renato Krohling

Federal University of Espírito
Santo - UFES Department of
Informatics,
PPGI, CT-VII
Av. Fernando Ferrari 514,
Campus de Goiabeiras
Vitória - ES,
CEP 29075-910, Brazil
krohling.renato@gmail.com

Po-Hsien Kuo

Department of Transportation
Studies,
Texas Southern University
3100 Cleburne Avenue,
Houston, TX 77004, United States
ckin44@gmail.com

Ying Li

Department of Transportation
Studies,
Texas Southern University
3100 Cleburne Avenue,
Houston, TX 77004,
United States
alee8291@gmail.com

Sajjad Ahmad Madani

Department of Computer Science,
CIIT
COMSATS, Abbottabad
Pakistan
madani@ciit.net.pk

Ricardo Martinez

Universidad Autónoma de Baja
California
Calzada Universidad 14418,
Parque Industrial Internacional
Tijuana
Tijuana B.C., 22390, Mexico
mc.ricardo.martinez@hotmail.com

Piotr Marusak

Institute of Control and
Computation Engineering,
Warsaw University of Technology
Nowowiejska 15/19
Warsaw, 00665, Poland
p.marusak@ia.pw.edu.pl

Masunori Matsuzaki

Graduate School of Medicine,
Yamaguchi University
2-14-1 Tokiwadai
Ube 753-8505, Japan

Patricia Melin

Tijuana
Institute of Technology
Tijuana, B.C.,
Mexico
pmelin@tectijuana.edu.mx

Fernando Mendes

IPC/I3N - Institute of Polymers
and Composites,
University of Minho
Campus de Azurem
Guimarães, 4800-058, Portugal
fmendes@dep.uminho.pt

Aleksander Mendyk

Department of Pharmaceutical
Technology and Biopharmaceutics,
Faculty of Pharmacy,
Medical College, Jagiellonian
University Medyczna 9 Street
30-688 Krakow, Poland
mfmendyk@cyf-kr.edu.pl

Marija Milanović

University of Belgrade,
Faculty of Mathematics
Studentski trg 16/IV,
Belgrade, 11000, Serbia,
marija.milanovic@gmail.com

Ali Moghimi

Ferdowsi University
Vakil Abad, Mashhad
Iran
moghimi@um.ac.ir

Javad Mohebbi

Islamic Azad University,
Department of Computer Science
4th Kilometer of
Mashhad-Quchan road
Quchan Branch, Iran
javad.mohebi@gmail.com

Oscar Montiel-Ross

Instituto Politecnico Nacional,
Centro de Investigacion y
Desarrollo de Tecnología
igital (CITEDI-IPN)
Av. del Parque No.1310,
Mesa de Otay, 22510,
Tijuana, B.C., Mexico
o.montiel@ieee.org

Tomoharu Nakashima

Department of Computer Science
and Intelligent Systems,
Osaka Prefecture University
Osaka, Japan
nakashi@cs.osakafu-u.ac.jp

Manoharan Narayanasamy

Sathyabama University
Jeppiaar Nagar, Rajiv Gandhi Road
Chennai-600119, India
indiamano123@yahoo.com

Mostafa Noruzi Nashalji

K. N. Toosi University of Technology
Seyyed Khandan Bridge
Tehran, Iran
mostafanoruzi@hotmail.com

Michele Ottomanelli

Technical University of Bari
Viale de Gasperi
Taranto 74100, Italy
m.ottomanelli@poliba.it

Kalyasundaram Perumal

Indira Gandhi Centre
for Atomic Research
Kalpakkam, Chennai-603102, India
kalyasundaram@igcar.gov.in

Sebastian Polak

Unit of Pharmacoepidemiology and
Pharmacoeconomics,
Faculty of Pharmacy,
Medical College, Jagiellonian
University Medyczna 9 Street
30-688 Krakow, Poland
spolak@cm-uj.krakow.pl

Mahdieh Poostchi

Iran University of Science
and Technology,
Computer Engineering Department
University Road,
Hengam Street, Resalat Square
Tehran, 16846-13114, Iran
ma_poostchi@comp.iust.ac.ir

Fengxiang Qiao

Department of Transportation
Studies,
Texas Southern University
3100 Cleburne Avenue,
Houston, TX 77004, United States
qiao_fg@tsu.edu

Lakshmesha Rao

Technische Universität
Kaiserslautern,
Integrierte Sensorsysteme
Erwin-Schrödinger-Strasse
Kaiserslautern, 67653, Germany

Ramadevi Rathinasabapathy

Sathyabama University
Jeppiaar Nagar, Rajiv Gandhi Road
Chennai-600119, India
rama_adarsh@rediffmail.com

Antonio Rodriguez

Universidad Autónoma de Baja
California
Calzada Universidad 14418,
Parque Industrial Internacional
Tijuana
Tijuana B.C., 22390, Mexico
ardiaz@uabc.mx

Mehdi Sadeghzadeh

Islamic Azad University,
Science and Research branch
Tehran 19585-466, Iran
m.sadeghzadeh@mahshahriau.ac.ir

Muhammad Sarfraz

Department of Information Science,
Kuwait University
Adailiyah Campus, P.O. Box 5969
Safat 13060, Kuwait
prof.m.sarfraz@gmail.com

Domenico Sassanelli

Technical University of Bari
Via Orabona, 4
Bari 70125, Italy
d.sassanelli@poliba.it

Gerald Schaefer

Department of Computer Science,
Loughborough University
Loughborough
LE11 3TU, UK
gerald.schaefer@ieee.org

Roberto Sepulveda

Instituto Politecnico Nacional,
Centro de Investigacion
y Desarrollo de Tecnología
Digital (CITEDI-IPN)
Av. del Parque No. 1310,
Mesa de Otay,
22510, Tijuana, B.C., Mexico
rsepulve@citedi.mx

D. Srikanth

IIT Kharagpur
Kharagpur 721 302
India
srikanth.147@gmail.com

Thomas Stützle

IRIDIA, CoDE,
Universite Libre
de Bruxelles, Brussels,
Belgium
stuetzle@ulb.ac.be

Noriaki Suetake

Graduate School of Science
Engineering,
Yamaguchi University
1677-1 Yoshida
Yamaguchi 753-8512, Japan
suetake@sci.yamaguchi-u.ac.jp

Roger Tait

Brain Mapping Unit,
University of Cambridge,
Cambridge
CB2 0SZ, UK
rt337@cam.ac.uk

Cristina Teixeira

IPC/I3N - Institute of Polymers
and Composites,
University of Minho
Campus de Azurem
Guimarães, 4800-058, Portugal
cteixeira@dep.uminho.pt

Mohammad Teshnehlab

K.N.Toosi University of Technology
Tehran 16315-1355, Iran
teshnehlab@eetd.kntu.ac.ir

Mohammad Teshnehlab

K.N.Toosi University of Technology
Tehran 16315-1355, Iran
teshnehlab@eetd.kntu.ac.ir

Ashutosh Tiwari

Cranfield University
Cranfield, Bedfordshire MK43 0AL
United Kingdom
a.tiwari@cranfield.ac.uk

Chris Turner

Cranfield University
Cranfield, Bedfordshire MK43 0AL
United Kingdom
c.j.turner@cranfield.ac.uk

Eiji Uchino

Graduate School of Science
and Engineering,
Yamaguchi University
1677-1 Yoshida
Yamaguchi 753-8512, Japan
uchino@yamaguchi-u.ac.jp

Prakash Vasudevan

Indira Gandhi Centre
for Atomic Research
Kalpakkam
Chennai-603102, India
prakash@igcar.gov.in

Maryam Vatankhah

Islamic Azad
University of Mashhad
Ostad Yosefi,
Ghasem Abad, Mashhad,
Iran
vatankhahmaryam@gmail.com

Kostas Vergidis

Cranfield University
Cranfield, Bedfordshire
MK43 0AL
United Kingdom

Armando Vieira

Department of Physics,
Instituto Superior
de Engenharia do Porto
R. S. Tomé
Porto, 4200, Portugal
asv@isep.ipp.pt

Barbara Wiśniewska

Unit of Pharmacoepidemiology
and Pharmacoconomics,
Faculty of Pharmacy,
Medical College,
Jagiellonian University
Medyczna 9 Street
30-688 Krakow, Poland
bbielska@cm-uj.krakow.pl

Zhengping Wu

University of Bridgeport
221 University Avenue
Bridgeport, CT, 06604, USA
zhengpiw@bridgeport.edu

Hao Wu

University of Bridgeport
221 University Avenue
Bridgeport, CT, 06604, USA
wuhao@bridgeport.edu

Lei Yu

Department of Transportation
Studies,
Texas Southern University
3100 Cleburne Avenue,
Houston, TX 77004, United States
yu_lx@tsu.edu

Qing Zhu

Department of Transportation
Studies,
Texas Southern University
3100 Cleburne Avenue,
Houston, TX 77004, United States
qingzhu0223@gmail.com

Jan Zidek

Brno University of Technology,
Institute of Materials Science
Purkynova 118
Brno, 612 00, Czech Republic
zidek@fch.vutbr.cz

Contents

Part I: Soft Computing for Modeling, Control, and Optimization

Optimization of Co-rotating Twin-Screw Extruders Using Pareto Local Search	3
<i>Cristina Teixeira, José Covas, Thomas Stützle, António Gaspar-Cunha</i>	
Efficient Predictive Control Algorithm Based on Fuzzy Hammerstein Models: A Case Study	11
<i>Piotr Marek Marusak</i>	
Selection of Points Inside Cutoff Radius by Scanning All Points Sorted in Memory	21
<i>Jan Zidek</i>	
Supervised Fuzzy Clustering Based Initialization of Fuzzy Partitions for Decision Tree Induction	31
<i>János Abonyi</i>	
Fault Detection of the Tennessee Eastman Process Using Improved PCA and Neural Classifier	41
<i>Mostafa Noruzi Nashalji, Mahdi Aliyari Shoorehdeli, Mohammad Teshnehlab</i>	

Part II: New Evolutionary Methodologies

Bare Bones Particle Swarm Applied to Parameter Estimation of Mixed Weibull Distribution	53
<i>Renato A. Krohling, Mauro Campos, Patrick Borges</i>	
Structural Optimization Using Harmony Search Algorithm	61
<i>D. Srikanth, S.V. Barai</i>	
Nash Dominance with Applications to Equilibrium Problems with Equilibrium Constraints	71
<i>Andrew Koh</i>	
A New Evolutionary Based Approach for Solving the Uncapacitated Multiple Allocation p-Hub Median Problem	81
<i>Marija Milanović</i>	

Part III: Soft Computing for Bioinformatics, Knowledge Management, and Data Mining

Prediction of the hERG Potassium Channel Inhibition Potential with Use of the Artificial Neural Networks	91
<i>Sebastian Polak, Barbara Wiśniowska, Malidi Ahamadi, Aleksander Mendyk</i>	
Michigan Style Fuzzy Classification for Gene Expression Analysis	101
<i>Gerald Schaefer, Tomoharu Nakashima</i>	
Feature Selection for Bankruptcy Prediction: A Multi-Objective Optimization Approach	109
<i>Fernando Mendes, João Duarte, Armando Vieira, António Gaspar-Cunha</i>	
Dynamic Time Warping of Segmented Time Series	117
<i>Zoltán Bankó, János Abonyi</i>	
Feature Selection Using Combine of Genetic Algorithm and Ant Colony Optimization	127
<i>Mehdi Sadeghzadeh, Mohammad Teshnehlab, Kambiz Badie</i>	

**Part IV: Soft Computing for Image and Signal Processing, and
Pattern Recognition**

Fully Automatic Boundary Extraction of Coronary Plaque in IVUS Image by Anisotropic Diffusion and T-S Type Fuzzy Inference	139
<i>Takanori Koga, Shohei Ichiyama, Eiji Uchino, Noriaki Suetake, Takafumi Hiro, Masunori Matsuzaki</i>	
Cost Effective 3D Localization for Low-Power WSN Based on Modified Sammon's Stress Function with Incomplete Distance Information	149
<i>Lakshmesha Rao, Kuncup Iswandy, Andreas König</i>	
Classification of Pressure Drop Devices of Proto Type Fast Breeder Reactor through Seven Layered Feed Forward Neural Network	157
<i>Ramadevi Rathinasabapathy, Sheela Rani Balasubramaniam, Manoharan Narayanasamy, Prakash Vasudevan, Kalyasundaram Perumal, Baldev Raj</i>	
Music Visualization by Means of Comb Filter and Relaxation Time According to Human Perception	165
<i>Mahdieh Poostchi, Iman Kamkar, Javad Mohebbi</i>	
GA/SVM for Diagnosis Sleep Stages Using Non-linear and Spectral Features	175
<i>Maryam Vatankhah, Mohammad Reza Akbarzadeh Totonchi, Ali Moghimi, Vahid Asadpour</i>	

**Part V: Special Session Soft Computing in Computer Graphics,
Imaging, and Vision**

Prostate Boundary Detection in Ultrasound Images Based on Type-II Fuzzy Sets and Modified Fuzzy C-Means	187
<i>Aboul Ella Hassanien, Gerald Schaefer, Hameed AlQaheri</i>	
Distributed Multimedia Computing Using Intelligent Agents	197
<i>Gerald Schaefer, Roger Tait</i>	
Object Recognition Using Simulated Evolution on Fourier Descriptors	205
<i>Muhammad Sarfraz, Jawad Usman Arshed, Sajjad Ahmad Madani, Muhammad Iqbal</i>	

Part VI: Special Session Soft Computing for Intelligent Control

- Embedding a KM Type Reducer for High Speed Fuzzy Controller into an FPGA** 217
Roberto Sepúlveda, Oscar Montiel-Ross, Oscar Castillo, Patricia Melin
- An Application of Fuzzy Lyapunov Synthesis in the Design of Type-2 Fuzzy Logic Controllers** 229
Nohe R. Cazarez-Castro, Luis T. Aguilar, Oscar Castillo
- Access Control and Resource Allocation for Peer-to-Peer Video Streaming Based on Fuzzy Logic** 237
Zhengping Wu, Hao Wu
- Bio-inspired Optimization Methods of Fuzzy Logic Controllers Applied to Linear Plants**..... 245
Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, Antonio Rodriguez

Part VII: Special Session Traffic and Transportation Systems

- An Adaptive Neuro-Fuzzy Inference System for Simulation of Pedestrians Behaviour at Unsignalized Roadway Crossings** 255
Michele Ottomanelli, Leonardo Caggiani, Giuseppe Iannucci, Domenico Sassanelli
- A Cellular Genetic Algorithm for Solving the Vehicle Routing Problem with Time Windows** 263
Iman Kamkar, Mahdieh Poostchi, Mohammad Reza Akbarzadeh Totonchi
- Fuzzy C-Means Based Stop Line and Crosswalk Placement at Right Turn Vehicle Exclusive Lane**..... 271
Fengxiang Qiao, Qing Zhu, Po-Hsien Kuo, Ying Li, Lei Yu

Part VIII: Surveys and Tutorials

- New Trends of Soft Computing Methods for Industrial and Biological Processes** 283
Bernardo Penna Resende de Carvalho, Leonardo Carneiro de Araújo

Evolutionary Multi-objective Optimisation of Business Processes	293
<i>Ashutosh Tiwari, Kostas Vergidis, Chris Turner</i>	
Computational Intelligence in Speech and Audio Processing: Recent Advances	303
<i>Aboul Ella Hassanien, Gerald Schaefer, Ashraf Darwish</i>	
Author Index	313
Subject Index	315

Part I

**Soft Computing for Modeling,
Control, and Optimization**

Optimization of Co-rotating Twin-Screw Extruders Using Pareto Local Search

Cristina Teixeira, José Covas, Thomas Stützle, and António Gaspar-Cunha

Abstract. A Pareto Local Search (PLS) algorithm was developed and applied to the screw configuration of co-rotating twin-screw extruders. This problem can be seen as a sequencing problem where a set of different screw elements are to be sequentially positioned along the screw in order to maximize the extruder performance. The results obtained were compared with previous results obtained with a Multi-Objective Evolutionary Algorithm (MOEA), which was previously developed by the authors. These results show that the PLS algorithm, despite its conceptual simplicity, is able to generate screws with good performance.

1 Introduction

Due to their geometrical flexibility and good mixing capacity, co-rotating twin screw extruders are widely used in the polymer compounding industry. This type of machines can easily be adapted to work with different polymeric systems, e.g., polymer blends, nanocomposites or highly filled polymers, taking into account its modular construction. However, this geometrical flexibility makes the performance of these machines strongly dependent on the screw configuration being used, i.e., defining the adequate screw geometry to use in a specific polymer system is an important process requirement. This can be seen as an optimization problem involving the selection of the location of a set of available screw elements along the screw axis.

In the case tackled in this paper, the optimization consists in permuting a specific number of different screw elements in order to maximize the global performance of

Cristina Teixeira · José Covas · António Gaspar-Cunha

IPC/I3N - Institute of Polymers and Composites, University of Minho, Guimarães, Portugal

e-mail: cteixeira, @dep.uminho.pt, jcovas, @dep.uminho.pt,

agc@dep.uminho.pt

Thomas. Stützle

IRIDIA, CoDE, Université Libre de Bruxelles, Brussels, Belgium

e-mail: stuetzle@ulb.ac.be

the system. This problem was designated as Twin Screw Configuration Problem (TSCP); it is a sequencing problem with, the aim of determining the position of screw elements along the screw axis. Since it involves several conflicting objectives, it is actually a multi-objective combinatorial optimization problem (MCOP).

The TSCP was previously tackled using a Multi-Objective Evolutionary Algorithm (MOEA) [1]. One important limitation of using MOEAs is the high number of necessary evaluations of the objective functions, since this implies running a numerical modelling routine, which has significant computational costs. Therefore, in the present work a Stochastic Local Search (SLS) algorithm was applied as an alternative for tackling this problem. In particular, we adopted the Pareto Local Search (PLS) strategy [2,3]. For that purpose, a detailed comparison to the previously designed MOEA using different objectives was made. In particular, this work is a part of a more comprehensive study where different approaches will be tested, including MOEA, Multi-Objective Ant Colony Optimization (MO-ACO), simple SLS algorithms and hybrids of these three classes of algorithms. The main motivation for the study of PLS here is its conceptual simplicity and its known very good performance.

This paper is organized as follows. In section 2, the twin-screw extrusion configuration problem is described. In section 3, the details of the algorithms used are discussed. Then, in section 4 the results are presented and discussed and we conclude in section 5.

2 Twin-Screw Extruders

Intermeshing co-rotating twin-screw extruders have by two Archimede-type screws with the same geometry rotating in the same direction inside a heated barrel [4]. The screws are usually built up by coupling individual screw elements with different geometries. Conveying, mixing and kneading elements are available, with distinct geometries. The performance of this type of extruders depends on the use of the correct sequence of elements, so that the extruder be able to accomplish its main functions, namely, transporting and melting the solid polymer, mixing and devolatilizing and forcing the polymer to pass through the die [4]. Polymer pellets or powder are usually fed inside the barrel at a pre-set feed rate. The rotation speed of the screws, together with the local temperatures and screw geometry subject the polymer to a variety of thermomechanical stresses along the screw axis.

Therefore, the co-rotating twin-screw configuration problem consists in defining the best location of a set of screw elements along the screw shaft as illustrated in figure 1. In this example the aim is to determine the position along the screw of 10 transport elements, 3 kneading blocks (with different staggering angles) and one reverse element.

The performance of each screw configuration is obtained by using an elaborated computer simulation of the polymer flow through the screw elements, taking into account the relevant thermal and rheological physical phenomena. The process comprises the following steps: (i) transport of solid material; (ii) melting; (iii) mixing and homogenisation; (iv) pressure generation and (v) flow through the die

[5, 6]. The flow characteristics are determined by the different geometries of the screw elements. Right handed elements have conveying properties while left handed and kneading blocks with a negative staggering angle create a flow restriction (generating pressure).

The computer simulation programme considers all above steps [6]. After the solid polymer is fed into the hopper, it will flow under starved conditions through transport elements. When a restrictive element is reached, the channel starts to fill up and the melting process is initiated. When the polymer is full melted, the flow develops with or without pressure in the remaining of the screw elements, depending on whether they are totally or partially filled; overall, the pressure is determined by the location of the restrictive elements. Each evaluation of a screw configuration takes about one to two minutes on current CPUs. Hence, the high computational effort required for these evaluations is an additional complicating factor and we require algorithms that use a low number of function evaluations.

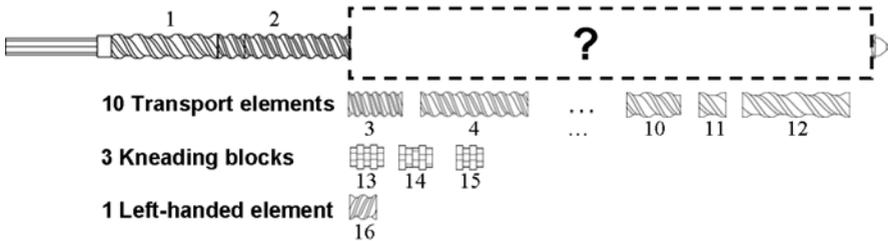


Fig. 1 Screw elements

3 Multi-objective Optimization

3.1 Multi-objective Evolutionary Algorithms

MOEAs have been recognized in the last decade as good methods to explore and find an approximation to the Pareto-optimal front for multi-objective optimization problems. This is due to the difficulty of traditional exact methods to solve this type of problems and by their capacity to explore and combine various solutions to find the Pareto front in a single run. A MOEA must provide a homogeneous distribution of the population along the Pareto frontier, together with an improvement of the solutions along successive generations [7, 8].

In this work, the Reduced Pareto Set Genetic Algorithm (RPSGA) is adopted [9, 10], where a clustering technique is applied to reduce the number of solutions on the efficient frontier. Initially, RPSGA sorts the individuals of the population in a number of pre-defined ranks using a clustering technique, in order to reduce the number of solutions on the efficient frontier, while maintaining its characteristics intact. Then, the individuals' fitness is calculated through a ranking function. To incorporate this technique, the traditional GA was modified [9, 10], following the

steps of a traditional GA, except for the existence of an external (elitist) population and for a specific fitness evaluation. Initially, an internal population of size N is randomly defined and an empty external population formed. At each generation a fixed number of best individuals, obtained by reducing the internal population with the clustering algorithm [10], are copied to an external population. This process is repeated until the number of individuals of the external population becomes full. Then, the RPSGA is applied to sort the individuals of the external population, and a pre-defined number of the best individuals are incorporated in the internal population by replacing low fitness individuals. Detailed information about this algorithm can be found elsewhere [9, 10].

3.2 Pareto Local Search Algorithm

SLS algorithms [11] have been successfully applied to single objective problems and, more recently, also to Multi-Objective Optimization Problems (MOOP). Successful single-objective based SLS algorithms can be readily extended to MOOPs via two strategies. The one we study here is to adopt a component-wise acceptance model, where a new solution is accepted in the local search if it is non-dominated by any of the other previous. As an example of such a strategy, we use Pareto Local Search (PLS) [2,3]. A key component of any local search algorithm is the definition of which solutions are neighbouring. Some preliminary experiments showed that the most suitable neighbourhood relation to be used in the local search is based on the 2-swap operator: two solutions are considered to be neighbours, if one can be obtained from the other by swapping the position of two screw elements.

The main ideas of PLS are the use of an archive, where all non-dominated solutions found so far are kept, and the exploration of the neighbourhood of each of these solutions using non-dominance criteria to decide about the acceptance of solutions [2,3]. The algorithm starts with a random initial solution. This is added to the archive and its neighbourhood is explored using the 2-swap operator. All non-dominated solutions identified in the neighbourhood exploration are added to the archive, if they are not dominated by any of the solutions in the archive, otherwise it is eliminated. These solution selection and archive update steps are iterated until the neighbourhood of all solutions in the archive has been explored. In order to avoid a too strong increase of the number of solutions in the archive, an archive bounding technique is used [12]. This bounding technique divides the objective space by a grid into hypercubes and allows only one non-dominated solution to occupy a given hypercube.

4 Results and Discussion

4.1 Case Study

The RPSGA and the PLS algorithms presented above were tested using the individual screw elements presented in Table 1 for a Leistritz LSM 30.34 twin-screw

extruder. In this example the objectives considered are the average strain, the specific mechanical energy (SME) and the viscous dissipation. Four instances and three different case studies were considered as presented, respectively, in Tables 1 and 2. Each optimization run was performed 10 times using different seed values. The comparison between the algorithms was made using the attainment functions methodology [13].

Table 1 Configuration of the individual screw elements for the 4 instances considered

Instance	Screw Element	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
TSCP1	Length (mm)	97.5	120	45	60	30	30	30	60	30	120	30	120	37.5	60	60	30
	Pitch (mm)	45	30	45	30	20	60	30	20	KB -60°	30	30	60	20	45	30	20
TSCP2	Length (mm)	97.5	120	45	60	30	30	30	60	30	120	30	120	37.5	60	60	30
	Pitch (mm)	45	30	45	30	-20	60	30	20	KB -60°	30	30	60	20	45	30	20
TSCP3	Length (mm)	97.5	120	45	60	30	30	30	60	30	120	30	120	37.5	60	60	30
	Pitch (mm)	45	30	KB -45°	30	-20	60	30	20	KB -60°	30	30	60	20	45	30	20
TSCP4	Length (mm)	97.5	120	45	60	30	30	30	60	30	120	30	120	37.5	60	60	30
	Pitch (mm)	45	30	KB -45°	30	-20	60	30	20	KB -60°	30	30	60	KB -30°	45	30	20

Table 2 Optimization objectives, aim of optimization and prescribed range of variation used in each case

	Objectives	Aim	X_{\min}	X_{\max}
Case study 1	Average Strain	Maximize	1000	15000
	Specific mechanical energy	Minimize	0.1	2
Case study 2	Average Strain	Maximize	1000	15000
	Viscous dissipation	Minimize	0.9	1.5
Case study 3	Specific mechanical energy	Minimize	0.1	2
	Viscous dissipation	Minimize	0.9	1.5

4.2 Comparative Results

In order to demonstrate the capacity of the PLS algorithm to deal with the TSCP, figure 2 shows a Pareto front for case study two and instance 1, considering a single run. As expected, the viscous dissipation (to be minimized) increases with the average strain (to be maximized). The viscous dissipation (measured as the ratio between the average melt temperature and the set barrel temperature) is smaller

when the restrictive elements are separated by conveying elements, since in this case the increase in temperature is also smaller. The opposite is true for the case of the average strain.

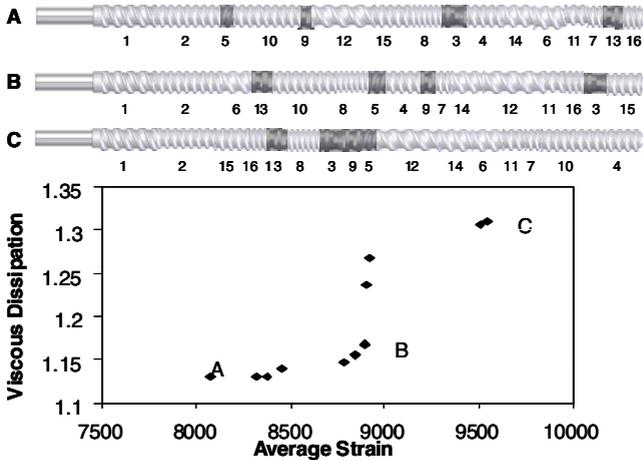


Fig. 2 Pareto front for case study 2, instance 1 and run 1 and some screw configurations generated

The results obtained with the PLS algorithm presented here were compared with the results obtained with the RPSGA algorithm developed previously. The comparison was made using the EAFs, as shown in Figure 3 for all cases studies and for instance 4. Similar results can be found on www.dep.uminho.pt/agc/results. The graphs on the left represent the region(s) of the Pareto front where the MOEA (in this case the RPSGA) is better, while the graphs on the right represent the region(s) where the PLS algorithm is better. The results shown in Figure 3 allow one to conclude that the PLS is better for instance 4. The same happens for instance 3, while for instance 1 and 2 the results are slightly better for the RPSGA algorithm.

5 Conclusions

A Pareto Local Search algorithm was applied with success to the Twin Screw Configuration Problem. The solutions obtained comply with the available scientific and technical knowledge on the process. The good performance obtained with the PLS algorithm alone is somewhat surprising, since it is a very simple method. In addition, the results indicate that it may be worthwhile to further develop the method, for example, by a better choice of which solutions are to be explored next, or to consider it for a possible combination with other methods, for example, as a post-processor to improve the search process of techniques such as the RPSGA.

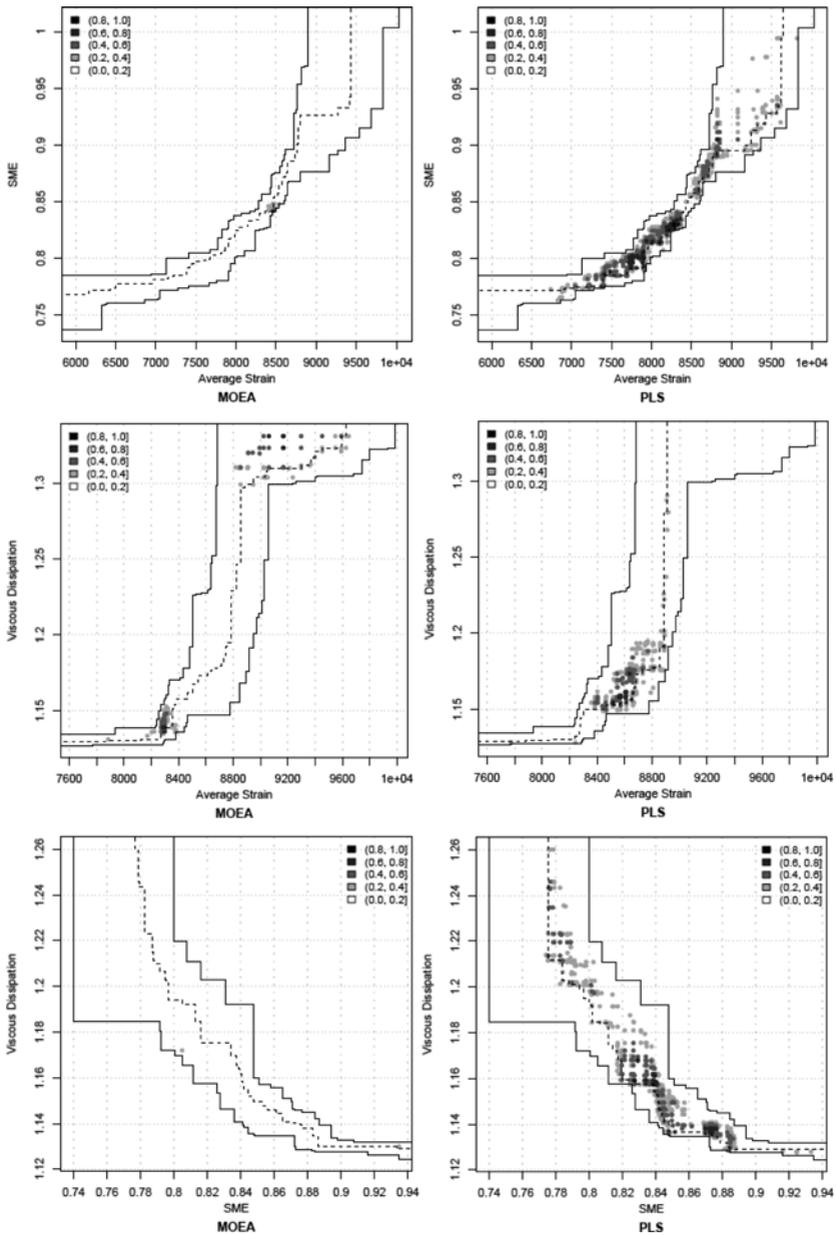


Fig. 3 EAFs differences between MOEA and PLS results for instance 4 and case studies 1 (top), 2 (middle) and 3 (bottom)

References

1. Gaspar-Cunha, A., Covas, J.A., Vergnes, B.: Defining the Configuration of Co-Rotating Twin-Screw Extruders With Multiobjective Evolutionary Algorithms. *Polymer Engineering and Science* 45, 1159–1173 (2005)
2. Paquete, L., Stützle, T.: A study of stochastic local search algorithms for the biobjective QAP with correlated flow matrices. *European J. of Operational Research* 169, 943–959 (2006)
3. Paquete, L., Chiarandini, M., Stützle, T.: Pareto local optimum sets in the biobjective traveling salesman problem: An experimental study. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) FAIR 1991. LNCS, vol. 535, pp. 177–200. Springer, Heidelberg (2004)
4. White, J.L.: *Twin Screw Extrusion; Technology and Principles*. Hanser, Munich (1990)
5. Vergnes, B., Della Valle, G., Delamare, L.: A Global Computer Software for Polymer Flows in Corotating Twin Screw Extruders. *Polymer Engineering and Science* 38, 1781–1792 (1998)
6. Teixeira, C., Gaspar-Cunha, A., Covas, J.A.: A Global Plasticating Software for Co-Rotating Twin Screw Extruders (Under preparation) (2009)
7. Gaspar-Cunha, A., Oliveira, P., Covas, J.A.: Use of Genetic Algorithms in Multicriteria Optimization to Solve Industrial Problems. In: *Seventh Int. Conf. on Genetic Algorithms*. Michigan, USA (1997)
8. Deb, K.: *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley, Chichester (2001)
9. Gaspar-Cunha, A., Covas, J.A.: RPSGAe - A Multiobjective Genetic Algorithm with Elitism: Application to Polymer Extrusion. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) *Metaheuristics for Multiobjective Optimisation*. LNCS, vol. 535, pp. 221–249. Springer, Heidelberg (2004)
10. Gaspar-Cunha, A.: *Modelling and Optimization of Single Screw Extrusion*. PhD Thesis, University of Minho, Guimarães, Portugal (2000), <http://www.lania.mx/~ccoello/EMOO/>
11. Hoos, H.H., Stützle, T.: *Stochastic Local Search - Foundations and Applications*. Morgan Kaufmann Publishers, San Francisco (2004)
12. Angel, E., Bampis, E., Gourvés, L.: A dynasearch neighborhood for the bicriteria traveling salesman problem. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) *Metaheuristics for Multiobjective Optimisation*. LNCS, vol. 535, pp. 153–176. Springer, Heidelberg (2004)
13. da Fonseca, V.G., Fonseca, C., Hall, A.: Inferential performance assessment of stochastic optimisers and the attainment function. In: Zitzler, E., Deb, K., Thiele, L., Coello Coello, C.A., Corne, D. (eds.) *EMO 2001*. LNCS, vol. 1993, pp. 213–225. Springer, Heidelberg (2001)

Efficient Predictive Control Algorithm Based on Fuzzy Hammerstein Models: A Case Study

Piotr Marek Marusak

Abstract. An efficient fuzzy predictive control algorithm based on Hammerstein models is proposed in the paper. The algorithm uses the DMC (Dynamic Matrix Control) technique and a Hammerstein model, in which fuzzy static block precedes a linear dynamic block. The static block may be identified easily using, e.g. heuristic approach and/or fuzzy neural networks. The dynamic part of the model has the form of control plant step responses. The proposed algorithm is little complicated and numerically effective (the main part of calculations is performed off-line) but it offers better control performance than a classical algorithm (based on a linear model). It is demonstrated in the example control system of a nonlinear control plant with significant delay.

1 Introduction

Model predictive control (MPC) algorithms are widely used in practice due to control performance they offer; see e.g. [2, 7, 13, 16]. Basic formulations of the MPC algorithms are based on the linear control plant models. However, application of such an MPC algorithm to a nonlinear plant may bring unsatisfactory results, especially if control in a wide range of set point values is needed.

Many processes, like for example distillation columns or chemical reactors can be successfully modeled by means of Hammerstein models, see e.g. [1, 5]. These models consist of the nonlinear static part preceding the linear dynamic part. The static part may be modeled using a fuzzy model [12, 14]. Thanks to such an approach one can obtain profits which are offered by the fuzzy approach (like e.g. relative easiness of model identification, relatively simple linearization, possibility to utilize the stability criteria developed for fuzzy systems). In the paper a Hammerstein model with fuzzy static part and the dynamic part in the form of the step responses of the control

Piotr Marek Marusak

Institute of Control and Computation Engineering, Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland

e-mail: P.Marusak@ia.pw.edu.pl

plant is used to synthesize the numerically efficient analytical nonlinear MPC controller. It is done in such a way that the main part of computations is performed off-line, like in the case of the classical analytical MPC controller. The proposed fuzzy controller outperforms the latter one, what is demonstrated in the control system of a distillation column.

The proposed approach has also advantages over the approach used e.g. in [1, 6] in which the inverse of the static part of the model is used. In some cases derivation of the inversed model may be impossible. Moreover, the mechanism of taking manipulated variable constraints into consideration is complicated there. It is because, after application of the inversed static model, the constraints put on manipulated variable are subject to a nonlinear transformation [1]. On the contrary, in the proposed approach, manipulated variable constraints can be handled in a straightforward way. It can be done either by using the control clipping mechanism [16] with the proposed analytical predictive controller or by using a numerical predictive algorithm (formulated as a quadratic optimization problem solved on-line at each sampling instant) being an adaptation of the one proposed in [8].

In the next section the idea of MPC algorithms and the analytical DMC algorithm are discussed. In Sect. 3 the motivating example is presented. In Sect. 4 the MPC controllers based on fuzzy Hammerstein models are proposed. Sect. 5 contains description of illustrative experiments. The paper is summarized in Sect. 6.

2 Predictive Control Algorithms

The MPC algorithms use a dynamic control plant model to predict future behavior of the control plant many sampling instants ahead. The future control values are derived in such a way that future behavior of the control system fulfills assumed criteria. Usually, the following performance function is minimized [3, 7, 13, 16]:

$$J_{MPC} = \sum_{i=1}^p (\bar{y}_k - y_{k+i|k})^2 + \sum_{i=0}^{s-1} \lambda \cdot (\Delta u_{k+i|k})^2 \quad (1)$$

where \bar{y}_k is a set-point value, $y_{k+i|k}$ is an output value for the $(k+i)^{\text{th}}$ sampling instant predicted at the k^{th} sampling instant using a control plant model, $\Delta u_{k+i|k}$ are future changes in the manipulated variables, $\lambda \geq 0$ is a weighting coefficient, p and s denote prediction and control horizons, respectively. After minimization of the performance function (1) the optimal vector of changes of the manipulated variables is obtained. From this vector, the $\Delta u_{k|k}$ element is applied in the control system. Then, optimization is repeated at the next sampling instant.

The way the predicted output values $y_{k+i|k}$ are derived depends on the dynamic control plant model exploited by the algorithm. If the linear model is used then the optimization problem with performance function (1) and without constraints has an analytical solution. In the case of the DMC algorithm it can be obtained as follows (the procedure is similar for other types of analytical MPC algorithms [16]).

The DMC algorithm is practically a standard in the industrial applications [2, 3, 4, 7, 13, 16]. It can be relatively easily designed, because it is based on an easy to obtain control plant model in the form of the step response:

$$\tilde{y}_k = \sum_{i=1}^{p_d-1} a_i \cdot \Delta u_{k-i} + a_{p_d} \cdot u_{k-p_d}, \quad (2)$$

where \tilde{y}_k is the output of the control plant model at the k^{th} sampling instant, Δu_k is a change of the manipulated variable at the k^{th} sampling instant, a_i ($i=1, \dots, p_d$) are step response coefficients of the control plant, p_d is equal to the number of sampling instants after which the coefficients of the step response can be assumed as settled, u_{k-p_d} is a value of the manipulated variable at the $(k-p_d)^{\text{th}}$ sampling instant.

The predicted output values are calculated using the following formula:

$$y_{k+i|k} = \sum_{n=1}^i a_n \cdot \Delta u_{k-n+i} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta u_{k-n+i} + a_{p_d} \cdot u_{k-p_d+i} + d_k, \quad (3)$$

where $d_k = y_k - \tilde{y}_{k-1}$ is assumed to be the same at each sampling instant in the prediction horizon (it is a DMC-type disturbance model). Thus (3) can be transformed into the following form:

$$y_{k+i|k} = y_k + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta u_{k-n+i} + a_{p_d} \cdot \sum_{n=p_d}^{p_d+i-1} \Delta u_{k-n+i} - \sum_{n=1}^{p_d-1} a_n \cdot \Delta u_{k-n} + \sum_{n=1}^i a_n \cdot \Delta u_{k-n+i|k}. \quad (4)$$

In (4) only the last component depends on future changes of the manipulated variable. Thus, the vector of predicted output values $\mathbf{y} = [y_{k+1|k}, \dots, y_{k+p|k}]^T$ can be decomposed into the following components:

$$\mathbf{y} = \tilde{\mathbf{y}} + \mathbf{A} \cdot \Delta \mathbf{u}, \quad (5)$$

where, $\Delta \mathbf{u} = [\Delta u_{k|k}, \dots, \Delta u_{k+s-1|k}]^T$, $\tilde{\mathbf{y}} = [\tilde{y}_{k+1|k}, \dots, \tilde{y}_{k+p|k}]^T$ is a vector of length p called a free response of the plant; it contains future output values calculated assuming that the control signal does not change in the prediction horizon (describes influence of the manipulated variable values applied to the control plant in previous sampling instants):

$$\tilde{\mathbf{y}} = \bar{\mathbf{y}} + \bar{\mathbf{A}} \cdot \Delta \bar{\mathbf{u}}, \quad (6)$$

where $\Delta \bar{\mathbf{u}} = [\Delta u_{k-1}, \dots, \Delta u_{k-p_d+1}]^T$, $\bar{\mathbf{y}} = [y_k, \dots, y_k]^T$ has p elements and

$$\bar{\mathbf{A}} = \begin{bmatrix} a_2 - a_1 & a_3 - a_2 & \cdots & a_{p_d-1} - a_{p_d-2} & a_{p_d} - a_{p_d-1} \\ a_3 - a_1 & a_4 - a_2 & \cdots & a_{p_d} - a_{p_d-2} & a_{p_d} - a_{p_d-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{p+1} - a_1 & a_{p+2} - a_2 & \cdots & a_{p_d} - a_{p_d-2} & a_{p_d} - a_{p_d-1} \end{bmatrix}. \quad (7)$$

\mathbf{A} is a $p \times s$ matrix, called the dynamic matrix and is composed of the control plant step response coefficients:

$$\mathbf{A} = \begin{bmatrix} a_1 & 0 & \cdots & 0 & 0 \\ a_2 & a_1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_p & a_{p-1} & \cdots & a_{p-s+2} & a_{p-s+1} \end{bmatrix}. \quad (8)$$

The solution of the minimization problem with performance function (1) without constraints can be calculated analytically using the following formula [16]:

$$\Delta \mathbf{u} = (\mathbf{A}^T \cdot \mathbf{A} + \lambda \cdot \mathbf{I})^{-1} \cdot \mathbf{A}^T \cdot (\bar{\mathbf{y}} - \tilde{\mathbf{y}}), \quad (9)$$

where \mathbf{I} is the identity matrix, $\bar{\mathbf{y}} = [\bar{y}_k, \dots, \bar{y}_k]^T$ is a vector of length p . From the vector $\Delta \mathbf{u}$ only the $\Delta u_{k|k}$ element must be calculated. Thus, only the first row of the matrix $(\mathbf{A}^T \cdot \mathbf{A} + \lambda \cdot \mathbf{I})^{-1} \cdot \mathbf{A}^T$ must be calculated and the control law can be easily obtained:

$$\Delta u_{k|k} = r_0 \cdot e_k - \sum_{i=1}^{p_d-1} r_i \cdot \Delta u_{k-i}, \quad (10)$$

where $e_k = \bar{y}_k - y_k$ is a control error at the k^{th} sampling instant, $r_0 = \sum_{j=1}^p K_{1j}$ $[r_1, \dots, r_{p_d}] = \mathbf{K}_1 \cdot \bar{\mathbf{A}}$, $\mathbf{K}_1 = [K_{11}, \dots, K_{1p}]$ is the first row of the matrix $\mathbf{K} = (\mathbf{A}^T \cdot \mathbf{A} + \lambda \cdot \mathbf{I})^{-1} \cdot \mathbf{A}^T$. The manipulated variable value is then calculated using the formula:

$$u_k = u_{k-1} + \Delta u_{k|k}, \quad (11)$$

where u_{k-1} is the manipulated variable value in the previous sampling instant.

3 Motivating Example

The control plant under consideration is an ethylene distillation column DA-303 from petrochemical plant in Plock. It is a highly nonlinear plant with a large time delay. The presented model of the column with basic feedback controllers was designed at the Institute of Control and Computation Engineering jointly with specialists from the Institute of Industrial Chemistry. It was assumed that the model is of Hammerstein type consisting of a nonlinear static block and a linear dynamic block (Fig. 1).

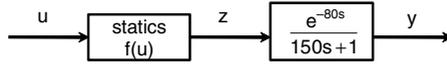


Fig. 1 Structure of the control plant model; time constants in minutes

The output of the plant y is the impurity of the product. From the economical efficiency point of view it should be as high as it is allowed. It is because in order to obtain better product, more energy must be supplied to the process. On the other hand, the allowable impurity cannot be exceeded because the product will be wasted, thus the responses without overshoot are preferred. The manipulated variable u is the reflux. The higher it is the purer product is obtained. During experiments it was assumed that the reflux is constrained $4.05 \leq u \leq 4.4$. These constraints were taken into consideration in the predictive controllers using a control clipping mechanism, see e.g. [16].

The static part of the model may have different forms. Using the set of samples the polynomial model was identified, as it is usually done when a Hammerstein model is used. However, the obtained model was very sensitive to the values of parameters – truncation of their values caused completely different model behaviour. Fortunately, using the fuzzy approach the approximation of the static characteristic of the plant can be obtained relatively easy using the expert knowledge and interaction with a computer. It was done so during this research. The Takagi–Sugeno model was used, with three antecedents of the form: $z_k^i = c_{1,i} \cdot u_k + c_{2,i}$, where $c_{11} = -2222.4$, $c_{12} = -1083.2$, $c_{13} = -534.4$, $c_{21} = 9486$, $c_{22} = 4709.3$, $c_{23} = 2408.7$. The assumed membership functions are shown in Fig. 2.

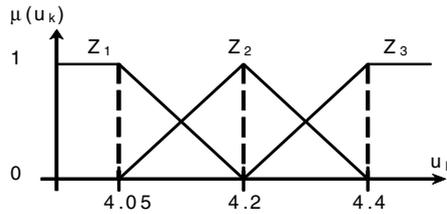


Fig. 2 Membership functions in the static part of the control plant model

There was made a test, how the classical DMC algorithm works for different set–points. The sampling time $T_s=20$ min, prediction horizon $p=44$ and control horizon $s=20$ were assumed. However, the controller designed for small set–point values causes instability for large set–point values (Fig. 3 left) and the controller designed to work well for large set–point values, works very slow for smaller set–point values (Fig. 3 right). The fuzzy controller should overcome these problems.

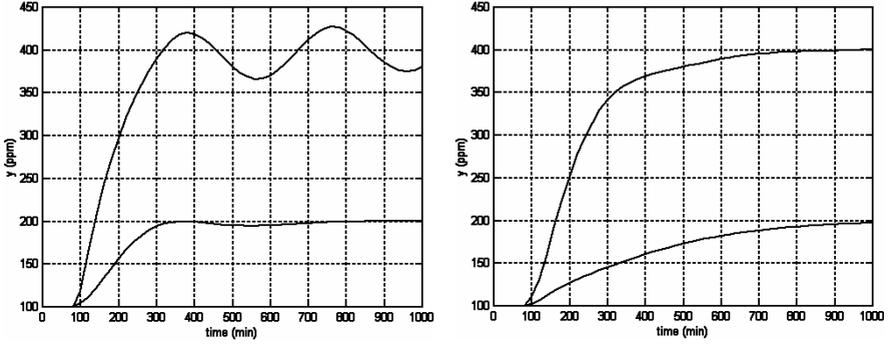


Fig. 3 Responses of the control systems with DMC algorithms: left – tuned to operate near 200 ppm, right – tuned to operate near 400 ppm

4 DMC Algorithm with Fuzzy Hammerstein Model

In order to design a fuzzy DMC controller the Hammerstein model with the dynamic part being the step response model is used. Next, at each iteration the following linear approximation of the model is obtained (using the fuzzy reasoning):

$$\tilde{y}_k = z_k \cdot \left(\sum_{i=1}^{p_d-1} a_i \cdot \Delta u_{k-i} + a_{p_d} \cdot u_{k-p_d} \right), \quad (12)$$

where $z_k = \sum_{i=1}^l \tilde{w}_i \cdot z_k^i$ is the output of the static part of the fuzzy Hammerstein model being the derivative or approximation of derivative of the steady-state characteristic of the process, \tilde{w}_i are normalized weights calculated using standard fuzzy reasoning, see e.g. [14], l is a number of fuzzy rules. It means that at each iteration the step response changes adapting to a nonlinearity of the control plant. Thus, one can use the methodology presented in Sect. 2 at each iteration of the controller (for each new set of step response coefficients). However, it would cause complication of the algorithm, because a matrix inverse must have been calculated on-line at each sampling instant.

The design process of the predictive controller may be, however, simplified to large extent thanks to exploitation of the form of the model (12). Let us notice, that at each iteration the dynamic matrix will now be as follows:

$$A_k = z_k \cdot A, \quad (13)$$

where A is a dynamic matrix given by (8) and the free response may be obtained using the following formula:

$$\tilde{y} = \bar{y} + z_k \cdot \bar{A} \cdot \Delta \bar{u}. \quad (14)$$

Remark 1. The free response can be also calculated using a nonlinear process model directly, see e.g. [9, 16]. Such an approach complicates the controller but may be useful for some control plants.

Let us assume that $\lambda_k = z_k^2 \cdot \lambda$. Therefore, after application of formula (9) at the current iteration one obtains:

$$\begin{aligned} \Delta \mathbf{u} &= (\mathbf{A}_k^T \cdot \mathbf{A}_k + \lambda_k \cdot \mathbf{I})^{-1} \cdot \mathbf{A}_k^T \cdot (\bar{\mathbf{y}} - \tilde{\mathbf{y}}) \\ &= (z_k^2 \cdot \mathbf{A}^T \cdot \mathbf{A} + z_k^2 \cdot \lambda \cdot \mathbf{I})^{-1} \cdot z_k \cdot \mathbf{A}^T \cdot (\bar{\mathbf{y}} - \tilde{\mathbf{y}}) \\ &= \frac{1}{z_k} \cdot (\mathbf{A}^T \cdot \mathbf{A} + \lambda \cdot \mathbf{I})^{-1} \cdot \mathbf{A}^T \cdot (\bar{\mathbf{y}} - \tilde{\mathbf{y}}). \end{aligned} \quad (15)$$

Thus, the most computationally demanding part of calculations (matrix inverse computation) can be still done off-line. Moreover, the control law of the fuzzy controller can be easily obtained, as in the linear case:

$$\Delta u_k = r_{0,k} \cdot e_k - \sum_{i=1}^{p_d-1} r_i \cdot \Delta u_{k-i}, \quad (16)$$

where $r_{0,k} = \frac{1}{z_k} \cdot r_0$ and r_0, r_1, \dots, r_{p_d} are calculated as in (10). Thus, the control law can be expressed as:

$$\Delta u_k = \frac{1}{\sum_{i=1}^l \tilde{w}_i \cdot z_k^i} r_0 \cdot e_k - \sum_{i=1}^{p_d-1} r_i \cdot \Delta u_{k-i}, \quad (17)$$

Remark 2. Thanks to the usage of the Hammerstein model with fuzzy static part (and, as a result, appropriate form of z_k) one can use Tanaka–Sugeno stability criterion [15] during the process of the controller design. In order to do so one can use similar adaptation of the Tanaka–Sugeno stability criterion as that presented in [10] for the analytical MPC algorithms. It can be done so because the normalized weights were used and the Tanaka–Sugeno criterion in its basic form does not depend on the shape of the membership functions.

4.1 Constraint Satisfaction Mechanisms

The manipulated variable constraints may be handled in the proposed algorithm in a straightforward way. Unlike in [1], a nonlinear transformation is not needed. As in the case of other analytical MPC algorithms a mechanism of control clipping can be used. This mechanism is relatively simple to implement and consists in application of the following rules [16]:

For constraints put on control changes:

- if $\Delta u_{klk} \leq \Delta u_{\min}$, then $\Delta u_{klk} = \Delta u_{\min}$,
- if $\Delta u_{klk} \geq \Delta u_{\max}$, then $\Delta u_{klk} = \Delta u_{\max}$;

For constraints put on control values:

- if $u_{k-1} + \Delta u_{k|k} \leq u_{\min}$, then $\Delta u_{k|k} = u_{\min} - u_{k-1}$,
- if $u_{k-1} + \Delta u_{k|k} \geq u_{\max}$, then $\Delta u_{k|k} = u_{\max} - u_{k-1}$,

where Δu_{\min} , Δu_{\max} , u_{\min} , u_{\max} are lower and upper limits of future changes and values of the manipulated variable, respectively.

The mechanism though simple is usually effective and gives satisfactory results [16]. However, if one needs, e.g. take output constraints into consideration, a numerical MPC algorithm can be used. Such an algorithm is formulated using the prediction which in the considered case is described by:

$$y = \tilde{y} + A_k \cdot \Delta u, \quad (18)$$

where A_k is given by (13) and the free response \tilde{y} is described by (14). Then a quadratic optimization problem, to be solved on-line at each sampling instant, is formulated. In this problem, in order to obtain the optimal Δu vector, the performance index (1) is minimized numerically, subject to the following constraints:

$$\Delta u_{\min} \leq \Delta u \leq \Delta u_{\max}, \quad u_{\min} \leq u \leq u_{\max}, \quad y_{\min} \leq y \leq y_{\max}, \quad (19)$$

where $u = [u_{k|k}, \dots, u_{k+s-1|k}]^T$, Δu_{\min} , Δu_{\max} , u_{\min} , u_{\max} , y_{\min} , y_{\max} are vectors of lower and upper limits of future changes and values of the manipulated variable and on predicted values of the output variable, respectively.

Remark 3. It is also possible to design a stable numerical MPC algorithm adapting the approach used in [11]. In such a case the analytical MPC algorithm may be used as the stabilizing controller.

5 Simulation Experiments

In order to improve control performance of the control system of the distillation column described in Sect. 3 a fuzzy DMC (FDMC) controller based on the fuzzy Hammerstein model with step response in the linear dynamic block was designed. In the static block the fuzzy model with membership functions shown in Fig 2 and three antecedents of the form: $z_k^i = c_{1,i}$, where $c_{11} = -2222.4$, $c_{12} = -1083.2$, $c_{13} = -534.4$ was used. The same sampling time and horizons were assumed as in the conventional DMC algorithm (Sect. 3). The value of the coefficient λ weighing components in the minimized performance function (1) was set to 8e6.

Responses obtained in the control system with the fuzzy predictive controller are shown in Fig. 4. This time, operation of the control system is satisfactory for all set-point values. Moreover, the character of the obtained responses is similar for different set-point values and there is no overshoot. Thus, the fuzzy controller offers better control performance than any of the DMC controllers based on a linear model.

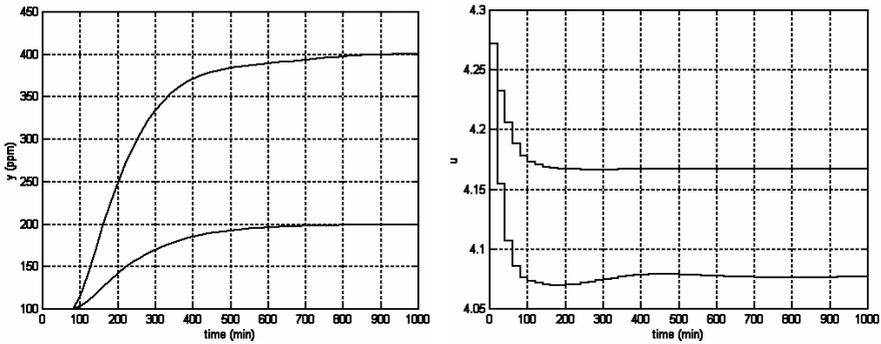


Fig. 4 Responses of the control system with FDMC algorithm: left – output signal, right – manipulated variable

6 Summary

The analytical fuzzy DMC (FDMC) algorithm based on a fuzzy Hammerstein model was proposed in the paper. It offers better control performance than the classical DMC algorithms based on linear models. At the same time, the FDMC algorithm has the same big advantage as its classical counterparts: on-line calculation of inversed matrix is avoided, thus, the main part of calculations is performed off-line.

Application of the proposed FDMC algorithm in the control system of the example nonlinear control plant with a large time delay brought improvement of the control system operation comparing to the case when the classical DMC controller was used. Thanks to the usage of the fuzzy approach, nominal stability of the control system can be easily tested during the design of the controller using Tanaka–Sugeno criterion.

Acknowledgement. The work was supported by the Polish national budget funds 2009–2011 for science as a research project.

References

1. Abonyi, J., Babuska, R., Ayala Botto, M., Szeifert, F., Nagy, L.: Identification and control of nonlinear systems using fuzzy Hammerstein models. *Industrial & Engineering Chemistry Research* 39, 4302–4314 (2000)
2. Blevins, T.L., McMillan, G.K., Wojsznis, W.K., Brown, M.W.: *Advanced Control Unleashed*. ISA – The Instrumentation, Systems, and Automation Society, Research Triangle Park (2003)
3. Camacho, E.F., Bordons, C.: *Model Predictive Control*. Springer, London (1999)
4. Cutler, C.R., Ramaker, B.L.: Dynamic Matrix Control – a computer control algorithm. In: *Proceedings of the Joint Automatic Control Conference*, San Francisco, CA, USA (1979)

5. Janczak, A.: Identification of nonlinear systems using neural networks and polynomial models: a block-oriented approach. Springer, Heidelberg (2005)
6. Jurado, F.: Predictive control of solid oxide fuel cells using fuzzy Hammerstein models. *Journal of Power Sources* 158, 245–253 (2006)
7. Maciejowski, J.M.: Predictive control with constraints. Prentice Hall, Harlow (2002)
8. Marusak, P.: Advantages of an easy to design fuzzy predictive algorithm in control systems of nonlinear chemical reactors. *Applied Soft Computing* 9, 1111–1125 (2009)
9. Marusak, P.: Efficient model predictive control algorithm with fuzzy approximations of nonlinear models. In: ICANNGA 2009, vol. 5495, pp. 448–457. Springer, Heidelberg (2009)
10. Marusak, P., Tatjewski, P.: Stability analysis of nonlinear control systems with unconstrained fuzzy predictive controllers. *Archives of Control Sciences* 12, 267–288 (2002)
11. Marusak, P., Tatjewski, P.: Effective dual-mode fuzzy DMC algorithms with on-line quadratic optimization and guaranteed stability. *International Journal of Applied Mathematics and Computer Science* 19, 127–141 (2009)
12. Piegat, A.: Fuzzy Modeling and Control. Physica-Verlag, Berlin (2001)
13. Rossiter, J.A.: Model-Based Predictive Control. CRC Press, Boca Raton (2003)
14. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its application to modeling and control. *IEEE Trans. Systems, Man and Cybernetics* 15, 116–132 (1985)
15. Tanaka, K., Sugeno, M.: Stability analysis and design of fuzzy control systems. *Fuzzy Sets and Systems* 45, 135–156 (1992)
16. Tatjewski, P.: Advanced Control of Industrial Processes, Structures and Algorithms. Springer, London (2007)

Selection of Points Inside Cutoff Radius by Scanning All Points Sorted in Memory

Jan Zidek

Abstract. A method for selection inside-cutoff points from a cloud of finite number of points was presented. The cutoff points are in less or equal distance from reference object than a given radius. This radius is user-defined input value. The proposed algorithm was inspired by a function of ribosome. Ribosome in living cells translates linear structure of ribonucleic acid to the complex structure of protein. In the virtual model of ribosome, similar functions were programmed. The linear structure of messenger-RNA was replaced by RAM. Programme imitated slider traversing the RAM. Inside the slider, there was linked list or binary search tree acting similarly like translation RNA in ribosome. The method applied to the searching the cutoff-points showed acceleration in comparison to classic methods.

1 Introduction

The inside-cutoff points detection is an algorithm applied to a cloud of finite number of points. In this cloud, the points in distance less or equal than in real software user-defined limit distance from given reference point were selected. The task is frequently more complicated by the multiple cutoff searching. That means that each point must be successively marked as reference and its points inside cutoff must be selected.

The algorithm of cutoff search (CS) is a part of class neighborhood analysis (NA). NA is analysis of local space distribution of data. The space distribution can be used in meaning of real geometrical objects [1] or virtual space, where the space is constructed from response of several sensor signals [2]. The tools used for NA are based on three algorithms. First is nearest neighbor search (NN). For the NN algorithm, advanced method k-dimensional tree [3] is presented. Second is a cluster analysis (CA). CA is performed by advanced algorithms DBSCAN [4] and

Jan Zidek

Institute of Materials Science, Brno University of Technology,

Purkynova 118, 61200 Brno, Czech Republic

e-mail: zidek@fch.vutbr.cz

OPTICS [5]. The third algorithm is a pair correlation function (PCF). The function value of PCF represents a point density distribution around a point.

The cutoff search algorithm is complementary to the other algorithms. It is used previously in molecular modeling chemistry [6, 7]. Selection of appropriate function from toolbox depends on required results. First aspect is nod-specificity of results. The specific result to concrete point is the matrix of neighbor points assigned to concrete central point (NN, CS). On the other hand, the non-specific result is a number or density of points characteristic for the cloud (PCF) or a cluster detected inside cloud (CA). The second aspect is local and global topological character of results. The typical local algorithm is CA, because more important points are inside cluster. The example of global function is PCF, which must be independent on selection of central point in system. Theoretically, PCF can be calculated up to infinite radius of sphere in the surrounding of central point, but in reality, it becomes constant above certain limit radius. This limit radius is threshold of macroscopic homogeneity. The systems above the limit radius can be considered macroscopically homogenous and composed by periodically repeated structure inside the limit radius. The CS and NN can be both local and global depending on system homogeneity given task.

Example of recent methods of NA is topological overlap measure approach (TOM and MTOM) [8]. The cloud of points is transformed to a network. The connection of two nodes is represented by value 1 disconnection by value 0. The single connections are performed to calculation of overlap functions. The distribution of overlap functions detects similarities of points.

The NA has many examples of applications. Specifically, the CS task is usually applied in software for molecular dynamics [6, 7]. It was applied also in AGLOMER software. It is software [1] for generation of finite set of non intersecting particles, which can represent for example the polymer filled with particles of mineral filler or space distribution of molecular coils. The CS method is processor time consuming and it is algorithm determining the runtime of entire program. Especially, when it must be done for each point in cloud and the program runs in multiple steps. Therefore, methods of acceleration are needed.

The classic methods are described in the literature for molecular dynamics software programming. A naïve method for this algorithm is comparison of distance of all pairs of points. Then the distances are compared to cutoff radius and classified to inside or outside cutoff region. This method is no more used. Some advanced methods were presented in literature [6]. One of the methods is for example cell subdivision. The space is divided into lattice of cubic cells, where each cell size must exceed the cutoff radius. The mutual distance is compared in the same cell and eventually in cells in close neighborhood.

The next method is a neighbor list search. Each point knows pointers or indices to its neighbors, which are in distance lower and equal to certain limit distance. The limit distance is larger than cutoff radius.

All the methods are computer-costs consuming, especially when in the atomic simulations number of atoms is more and more increasing. For example, there is a demand for molecular description of spherulites crystallization, where spherulite is spherical polyethylene crystal. The molecular size of the smallest possible

spherulite of polyethylene is in order of millions atoms. Therefore, the crystallization must be in this time presented only by supermolecular models [9]. The algorithm, which could solve this task, must have increased efficiency mainly in case of large number of points and low relative cutoff radius to size of cloud. It is common situation in real systems, because the cutoff radius is given as a range of molecular interactions. It is numerically constant and independent on size of system. Relative range of molecular interaction to the size of system is then decreasing when increasing atom numbers.

The proposed algorithm can process the cloud with large number of points is presented. It is inspired by function of ribosome. The ribosome is a complex structure inside living cells [10]. Its function is a translation of genetic code from RNA to the protein structure. The function of ribosome is shown in *Figure 1*. It can be in certain circumstances compared to the computer function with input and output data. The requirements to the ribosome functions are really exacting. The process in live cell must be quick and must warrant error-free run, because the protein structure influences some vital functions.

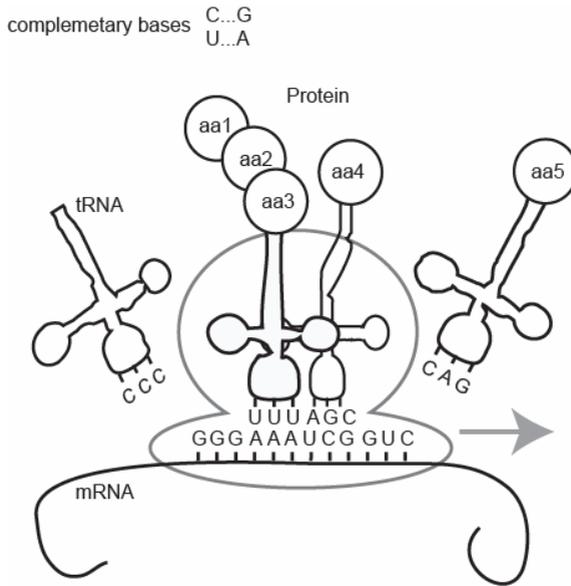


Fig. 1 Function of ribosome; ribosome – grey outline; mRNA – messenger ribonucleic acid; tRNA transfer RNA; aa – amino-acid; C, G, A, U – RNA bases: cytosine, guanine, adenine and uracil respectively U is complementary to A; C to G;

A protein must be compounded of correct amino acids. They must be moreover connected in correct sequence and geometric order. For this purpose, ribosome provides the process called translation. In real bio-systems the translation consists of seven steps [10] and some additional regulation mechanisms. In first approximation,

the process can be divided to two phases. The phases can be considered subroutines of certain function. The first is application of messenger-RNA (mRNA) which is information ordered to linear array of data. The second is linking of transfer-RNA (tRNA) which is selection of correct amino acid and its correct orientation in a protein. The ribosome is an object which traverses the linear mRNA and provides linking the correct amino acid for each “word” of genetic information.

2 Method

The details of the process inside ribosome can be compared to a function of cutoff region. The proposed software is inspired by ribosome, but does not copy its function absolutely. The first difference is that the result of programme is the set of selected points independently on their sequence in system.

Next difference is that the ribosome uses two sets: one set of input data (RNA bases) and another set of output data (amino acids). The virtual task is designed so the input (reference points) and output (inside cutoff points) were from the same data set. The data set is a cloud of all points. The virtual ribosome, which is adapted to the task, is shown in *Figure 2*.

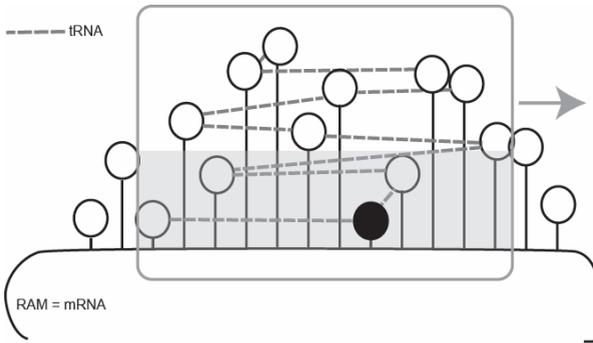


Fig. 2 Virtual ribosome adapted to problem of cutoff search; grey outline – virtual ribosome, circles – set of investigated points; black disk – reference point; grey dashed – translation RNA; grey surface – points inside cutoff region

The points in cloud represented by circles in *Figure 2* are related to reference point (black disc). The linear mRNA can be replaced by array in RAM with object sorted by certain property (for example one coordinate). The tRNA can be symbolized by pointer to some dynamically allocated object (DAO). The object detects the points in neighborhood of reference point. The DAO can be arranged for example linked list or binary search tree structure. When analyzing next points as reference, the most of structures are conserved and are only modified.

The application can be demonstrated at the example which was concretely implemented to software AGLOMER [1]. This is a demonstration of adaptability of the algorithm to the specific type of task. In an original algorithm the data are

analyzed by traversing points sorted by one coordinate. In the AGLOMER software, the situation is more complicated. There is self organization method providing random but non-intersecting distribution of spheres in space. The self organization method proceeded from center of mass to the boundary of cloud. Therefore, the points were sorted in RAM according to the distance from mass center. The data were linearly arranged to the memory (see *Figure 3 a*) similarly to the classic algorithm. However, their space distribution the data reminds a spiral. The ribosome processes the data like a gramophone needle (see *Figure 3 b*). The data inside ribosome were delimited by a segment of spiral, which is delimited by upper stop 1 and downer stop 1 (see *Figure 3 c*). Then the (x, y) coordinates of points from *Figure 3c* were projected to one (x) coordinate sorted to linked list (*Figure 3 d*). Structure was delimited by downer and upper stop No 2. The walking inside a linked list around the reference point was performed until the distance of cutoff radius (r).

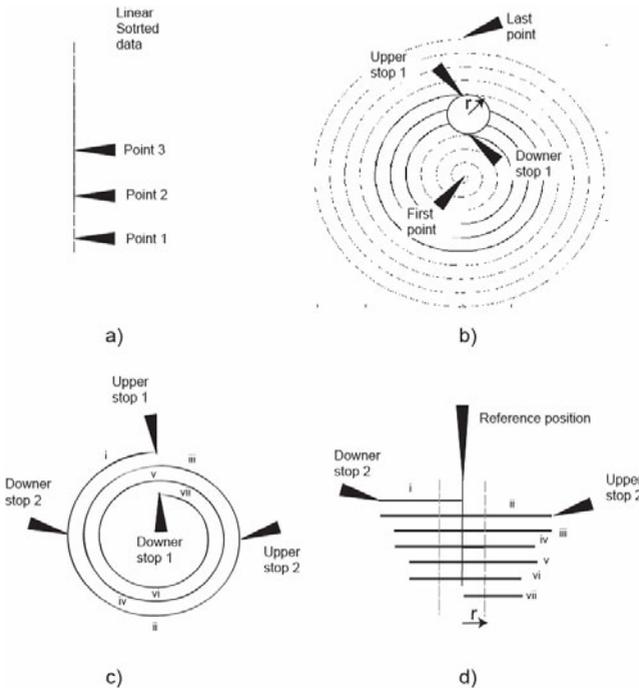


Fig. 3 Scheme of method applied in AGLOMER software; a) Sorting of points to the block of memory; b) traversing the data by ribosome: projection to x, y -plane, c) selection of (x, y) data inside ribosome; d) walking inside linked list in selected data until reaching boundary of cutoff region (r): x projection from x, y plane in c).

In the final set, there are among correct points also several points more which do not satisfy the cutoff condition. The points in this set must be still analyzed classically by the calculation the distance from reference point. However, the

calculation of distance function was rapidly decreased in comparison with standard methods.

The process had advantage when there are analyzed inside cutoff of every point in a set. It can be demonstrated in *Figure 4*. In this case, first step of “translation procedure” is only shifting the upper and downer stop No 1. The linked list is conserved and only the elements which arrived into the linked list must be sorted and the elements which leaved linked list must be deleted.

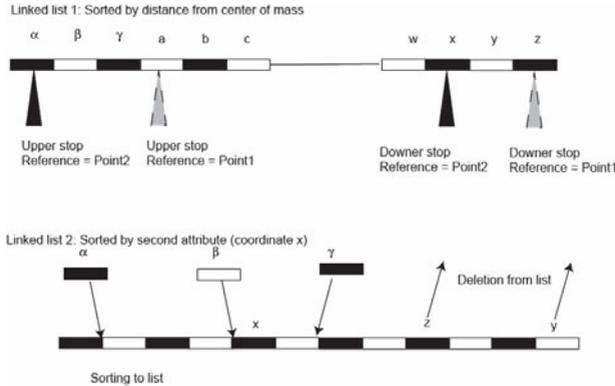


Fig. 4 Transition of analysis from one reference point to the next

3 Results and Discussions

The software was analyzed by two ways: balance of operations and total runtime of method performed on testing software.

The basic operations were classified to three groups: (i) condition: very quick operation; (ii) standard operation with mean time consumption and (iii.) square root which is relatively slow.

The basic operations were applied to the orientation estimation of complex operations.

First, the complex operations were calculated from basic operations. It was found that the distance of points was driving operation in the all-pairs and box subdivision method. It is composed of six subtractions, three multiplications and two additions, one square root and one condition. The balance of operations is then $[\text{conditions/standard/sqrt}] = [1/11/1]$.

In box subdivision, there were also additional operations. They were two: 1. detection whether both points are in the same box, and 2. operation sorting the points into the appropriate boxes. Their influence to the runtime was negligible in comparison to the distance calculation.

In the new method (ribosome), there was applied operation of sorting the element to the linked list. It was performed by browsing the linked list from the first, the last, or reference element. There was necessary to pass on average $n_{LL}/4$ steps, to find the correct positions, where n_{LL} was a number of segments inside linked

list (ribosome). The next operation in ribosome was walking inside linked list. Its runtime was negligible in comparison to building of linked list.

The methods are summarized in *Table 1*. All pairs comparison and box subdivision had the main method calculation of distance, which is relatively long due to long basic operations square root. Moreover its costs are $O(n^2)$ in the algorithms. The ribosome method had a driving operation: building linked list whose costs are $O(n \cdot n_{LL})$, where $n_{LL} \ll n$.

The numerical balance of operations is shown in *Table 1*. It was made for a box size =1 which contained 10^3 - 10^6 points. The sub-box size for box subdivision was $b = 0.1$ and the cutoff radius was 0.001. The condition was considered 50 times quicker than standard operation and square root 50 times slower.

Table 1 Number of operations consumed for detection of inside-cutoff points; two standard and one new method; n –number of points inside cloud; n_{LL} – number of points inside linked list (ribosome)

	Complex operations:	Costs per operation	Estimated runtime
All pairs	Distance of points	$O(n^2)$	1
Box subdivision	Detection of points in same box	Negligible	0.1-0.7
	Assignment points to box	Negligible	
The proposed method	Sorting element into linked list	$O(n \cdot n_{LL})$, $n_{LL} \ll n$.	0.001
	Walking inside linked list	Negligible	

The times calculated from operations balance was stated as relative to all pair methods. Of course, the numbers shown in *Table 1* are rather qualitative. It was only a demonstration that it is worthy to develop the new method. The real time consumption must be measured directly in the programme.

The test software was programmed in C++ and compiled by GNU compiler MinGW/GCC 3.3.1. The hardware facilities were 2x Dual core AMD OPTERON 265 1.79 GHz; 2GB DDR; MS Windows x64 SP2. Only system processes were running together with software.

The measured time interval was bounded exactly by beginning and ending of function. The cloud was composed of 200000 points with random position inside box of size 1x1x1. Cutoff radius was 0.001 of box side. Mean value and standard deviation was calculated from 10 independent runs of programme. Runtime of all pairs method was 4349 ± 3 s, box subdivision method 827 ± 2 and newly proposed method took 21 ± 3 s. When changing the input parameters, the relative times are similar.

It can be seen, than the new method is relatively accelerated in comparison to standard methods. However in acceleration of method in this stage are still the reserves. They are in three groups.

(i) First reserve is in using binary search tree (BST) instead of linked list. The BST reduces the disadvantage of linked list in searching correct position for new record. It leads to strong acceleration, especially when there is large number of points in the system. It was just programmed but is still being tested, in this time.

(ii.) Second reserve is introduction of basic principles of software acceleration. For example in this case, both the distances and cutoff radius are positive numbers. Therefore, comparison of distances and cutoff will have the same logical value as the comparison its squares of distance. So, the time-consuming calculation of square roots can be reduced. This was not introduced to the testing software, because there was investigated only effect of method. However, various basic principles of acceleration are known very well and can be always introduced to the real software according to concrete situation.

(iii) Replacement of distance calculation by condition and subtraction. In case the points are too distant, the distance in one coordinate excludes reliably the presence inside cutoff radius. Then number of operations is reduced.

It might be still one reason why the imitation of ribosome can be noticeable. In my opinion, the understanding of ribosome function can be helpful help in cryptology. In a genetic code of protein, there are only 4 letters, producing 64 words and the proteins are formed by 20 building blocks (amino acids). This small number of elements can code billions of proteins exist in such diversified structures as enzymes, collagen, muscular proteins (myosins) or for example spider silk.

4 Conclusion

The detection of points inside cutoff radius is relatively frequent problem. Some advanced methods were presented in literature but they were not sufficient to process large set of points. A new function imitating the function of ribosome was presented in this paper. The algorithm scans RAM similarly like ribosome messenger RNA. The next component – tRNA can be symbolized by various data structures. In this case, the tRNA in real ribosome was replaced by a simple linked list in virtual model. The processor-time of algorithm decreased orderly in comparison to the standard methods. The method can be even more accelerated in future by replacement of some components by others or by addition of standard principles of computer acceleration. The imitation of function of ribosome might be also applied in other fields of computing.

Acknowledgments. This research was supported by Grant Agency of The Czech Republic under the project GA106081409 and by Ministry of Education of the Czech Republic under research project MSM 0021630501.

References

- [1] Zidek, J., Kucera, J., Jancar, J.: Model of Randomly distributed set of nonintersecting particles with controlled macroscopic homogeneity. Aglomer software webpage, <http://aglomer.webnode.cz>

- [2] He, W., Hua, X., Zhao, L., Liao, X., Zhang, Y., Zhang, M., Wu, J.: Food Research International 42, 1462–1467 (2009)
- [3] Moore, A.W.: An introductory tutorial on kd trees, extracted from PhD. Thesis Efficient Memory based Learning for Robot Control, Computer Laboratory, University of Cambridge (1991)
- [4] Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (1996)
- [5] Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J.: OPTICS: Ordering Points To Identify the Clustering Structure. In: ACM SIGMOD international conference on Management of data, pp. 49–60. ACM Press, New York (1999)
- [6] Rapaport, D.C.: The art of Molecular Dynamics Simulation, 2nd edn. Cambridge University Press, Cambridge (2004)
- [7] Rodrigues, C.I., Hardy, D.J., Stone, J.E., Schulten, K., Hwu, W.-M.W.: GPU Acceleration of Cutoff Pair Potentials for Molecular Modeling Applications. In: Proceedings of the 5th conference on Computing frontiers, Ischia, Italy, pp. 273–282 (2008)
- [8] Li, A., Horvath, S.: Network neighborhood analysis with the multinode topological overlap measure. Bioinformatics 23, 222–231 (2007)
- [9] Toda, A., Okamura, M., Taguchi, K., Hikosaka, M., Kajioaka, H.: Branching and Higher Order Structure in Banded Polyethylene Spherulites. Macromolecules 41, 2484–2493 (2008)
- [10] Beier, W.: Biofysika Academia, 1974, Translated from: Biophysik. 3rd edn. Georg Thieme Leipzig (1968)

Supervised Fuzzy Clustering Based Initialization of Fuzzy Partitions for Decision Tree Induction

János Abonyi

Abstract. The generation of highly precise and interpretable models is the most frequent task of data mining. The interpretability of models and the nature of learning algorithms require effectively discretized (partitioned) features. Fuzzy partitioning of continuous features can increase the flexibility of the classifier since it has the ability to model fine knowledge details. Decision tree and rule induction methods often relay on *a priori* discretized (partitioned) continuous features. However, advantages of the supervised and fuzzy discretization of attributes have not yet been shown. This paper includes such study and proposes supervised clustering algorithm for providing informative input information for decision tree induction algorithms.

1 Introduction

As a result of the increasing complexity of classification problems it becomes necessary to deal with structural issues of the identification of classifier systems. The input attributes of a classification problem can be basically continuous or discrete in values. Also the interpretability of models and the nature of some learning algorithms require effectively discretized (partitioned) features. Hence, the effective partitioning of input domains of continuous input variables is an important aspect of the accuracy and transparency of classifier systems. The discretization can be *a priori* or dynamic. In *a priori* partition methods the attributes are partitioned before to the induction of the tree. Contrarily, dynamic methods generate the partitions during the tree induction mechanism. The most known decision tree induction methods that utilize *a priori* partitions are the ID3 [9] and the AQ [7] algorithms. The most current dynamic methods are the CART [2], the Dynamic-ID3 [4], the Assistant and the C4.5 algorithms.

János Abonyi

Department of Process Engineering, University of Pannonia, Veszprem P.O.Box 158, H-8201 Hungary

e-mail: abonyij@fmt.uni-pannon.hu

www.fmt.uni-pannon.hu/softcomp

Fuzzy partitioning of continuous features can increase the flexibility of the classifier since it has the ability to model fine knowledge details. (Fuzzy) decision trees based on the original ID3 algorithm assume discrete domains with small cardinalities. Another important feature of ID3 trees is that each attribute can provide at most one condition on a given path. These are great advantages as they increase comprehensibility of the induced knowledge, but may require a good *a priori* partitioning, because the first discretization step of the data mining procedure especially affects the classification performance. Hence, for this purpose Janikow introduced a genetic algorithm to optimize this partition step [5], while Pedrycz proposed an environment dependent clustering algorithm to get adequate partitions [8].

As these papers illustrate decision tree and rule induction methods often rely on *a priori* discretized (partitioned) continuous features. However, the advantages of the supervised and fuzzy discretization of the attributes have not yet been shown. This paper includes such study and proposes a new supervised clustering algorithm for providing informative input information for decision tree induction algorithms. The FID algorithm is applied for the induction of the decision tree [6]. This method has the advantage that the resulted fuzzy decision tree can be transformed into a fuzzy rule based classifier and the classification performance and the complexity of this model can fine-tuned by a rule pruning algorithm. Therefore, as the experimental study in this paper will illustrate, compact, perspicuous and accurate fuzzy classifier models can be identified by this approach.

2 Fuzzy Decision Tree Classifiers

Decision tree induction is the most important discriminative learning algorithm working by recursive partitioning. The basic idea is to partition the sample space in a data-driven manner, and represent the partition as a tree.

Fig. 1 shows a fuzzy decision tree and the fuzzy partitions of its attributes. This tree can be represented in terms of **If-Then** logical rules, where each concept is represented by one disjunctive normal form, and where the antecedent consists of a sequence of attribute value tests (e.g. If x_3 is large and x_2 is medium and ...). These attribute value tests partition the input domains of the classifier into intervals represented by the fuzzy sets, and the operating region of the rules is formulated by *And* connective of these domains. One of the key idea of this paper is to define the consequent of the rules as the probabilities of the given rule represents the c_1, \dots, c_C classes ($P(c_1|r_i), \dots, P(c_C|r_i)$), so that fuzzy decision trees obtained by the FID algorithm can be represented by an extended fuzzy classifier proposed in [1]. Therefore the proposed form of the classification rules is the following:

$$r_i : \mathbf{If } x_1 \text{ is } A_{i,1}(x_{1,k}) \mathbf{ And } \dots x_n \text{ is } A_{i,n}(x_{n,k}) \mathbf{ Then} \quad (1)$$

$$\hat{y}_k = c_1 \text{ with } P(c_1|r_i) \dots, \hat{y}_k = c_C \text{ with } P(c_C|r_i)$$

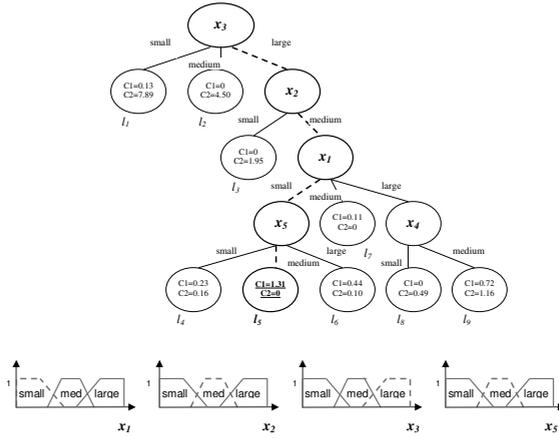


Fig. 1 A fuzzy decision tree and rules as partitions of attributes

where i is the index of the rule ($i = 1, 2, \dots, R$) and $\mathbf{x}_k = [x_{1,k}, x_{2,k}, \dots, x_{n,k}]^T$ is the input (features, attributes) vector, where $k = 1, \dots, N$ the index of data, n denotes the number of inputs and the $A_{i,1}, \dots, A_{i,n}$ represent the fuzzy sets in the antecedent part of the i th rule. The **And** connective represents the product operator between fuzzy sets, so the activity of the i th rule is calculated in the following way $\beta_i(\mathbf{x}) = \prod_{j=1}^n A_{i,j}(x_j)$, $i = 1, 2, \dots, R$. The output of the fuzzy rule base classifier is determined by the *winner takes all* strategy, by the aggregation of rules with normalized fuzzy mean formula. Accordingly for a given data sample (\mathbf{x}_k) the class label which has the highest rule activities will be the measured output (\hat{y}_k):

$$\hat{y}_k = c_{i^*}, i^* = \max_{1 \leq i \leq C} \frac{\sum_{l=1}^R \beta_l(\mathbf{x}_k) P(c_i | r_l)}{\sum_{l=1}^R \beta_l(\mathbf{x}_k)} \quad (2)$$

The presented fuzzy classifier is an extension of the quadratic Bayes classifier which uses a mixture of models to estimate the conditional densities of the classes:

$$\mathbf{x} \text{ is assigned to } c_i \iff \sum_{l=1}^R p(\mathbf{x}|r_l)P(r_l)P(c_i|r_l) \geq \sum_{l=1}^R p(\mathbf{x}|r_l)P(r_l)P(c_j|r_l) \forall j \neq i \quad (3)$$

where the $p(\mathbf{x}|r_l)$ distribution represents how the antecedent membership functions partition of the input variables of the classifier. For more details about this interpretation can be found in the Appendix where a supervised clustering algorithm is presented that is able to identify this kind of classifier model.

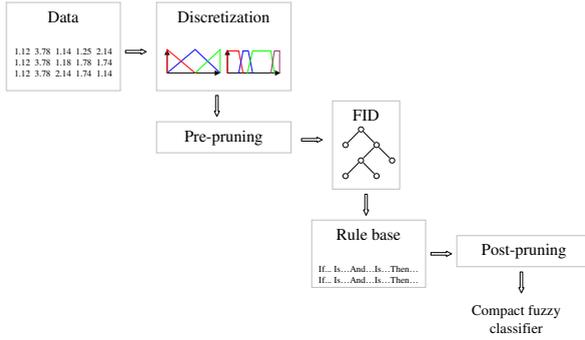


Fig. 2 The proposed method

3 Supervised Clustering Based Discretization of Attributes

The problem that is addressed in this paper is how fuzzy sets can be designed to provide effective *a priori* partitioning of the input variables for fuzzy decision tree induction. The most frequently applied method is the Ruspini-type partitioning where triangular membership functions are used [10]. This method resembles to classical equidistant discretization, since it does not utilize information inherent in data (e.g. class labels, distribution of the variables).

Clustering algorithms are often used to estimate the distribution of the data, hence they are useful for the discretization of continuous features [3]. Since these algorithms do not utilize class labels, the partitions (fuzzy sets) extracted from the clusters are not tailored to the classification problem. This unsupervised approach can be considered as equal frequency discretization. The key idea of this paper is to use supervised fuzzy clustering algorithm presented in [1] for the global, supervised, and fuzzy discretization of the features. The whole methodology is the following (see fig. 2).

Feature Discretization by Supervised Fuzzy Clustering: The first step of the proposed methodology is the supervised clustering of the available data. The clustering algorithm described in [1] allows the direct identification of fuzzy classifiers. Hence, the performance of these "draft" rule-based classifiers can also be evaluated. Since the supervised clustering algorithm uses Gaussian membership functions (fig. 3) and FID needs triangular or trapezoidal membership functions, the resulted Gaussian membership functions have to be transformed into trapezoids. The four characteristic points of the trapezoids are calculated as: $a_{i,j} = v_{i,j} - 3\sigma_{i,j}$, $b_{i,j} = v_{i,j} - 0.5\sigma_{i,j}$, $c_{i,j} = v_{i,j} + 0.5\sigma_{i,j}$, $d_{i,j} = v_{i,j} + 3\sigma_{i,j}$. **Pre-pruning:** The obtained fuzzy sets often significantly overlap, hence the resulted fuzzy classifier is often more complex than needed. To obtain a parsimonious and interpretable fuzzy classifier a fuzzy set-merging algorithm can be applied [11]. A similarity measure (S)

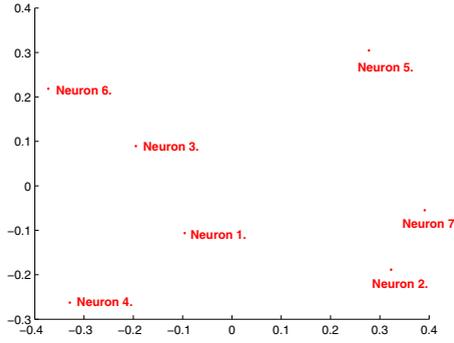


Fig. 3 Attribute partitioning by supervised fuzzy clustering

based on the set-theoretic operations of intersection and union is used to quantify the overlap of the membership functions:

$$S(A_{i,j}, A_{k,j}) = \frac{|A_{i,j} \cap A_{k,j}|}{|A_{i,j} \cup A_{k,j}|} \quad (4)$$

where $|\cdot|$ denotes the cardinality of a set, and the \cap and the \cup operators represent the intersection and union, respectively. If $S(A_{i,j}, A_{k,j}) = 1$, then the two membership functions ($A_{i,j}$ and $A_{k,j}$) are identical. $S(A_{i,j}, A_{k,j})$ becomes zero when the membership functions are non-overlapping. The overlapping fuzzy sets are merged when their similarity exceeds a user-defined threshold $\theta \in [0, 1]$ ($\theta = 0.8$ in this study). This merging reduces the number of fuzzy sets and thereby increases the transparency of the resulted classifier. The similarity measure is also used to detect *don't care* variables. If the merging procedure leads to only one membership function for a given variable, then this input variable can be eliminated [11]. **Tree induction:** The determined fuzzy membership functions are used as a priori partitions of the input variables by FID fuzzy decision tree induction algorithm proposed by Janikov [6]. The results obtained by this classifier will be denoted by FID in the experimental session. **Fuzzy rule base:** The resulted fuzzy decision tree has the important property that the leaves store the possibilities of all the classes. Therefore the presented interpretation of the decision tree allows its transformation into a rule base (fuzzy) system:

- **Step 0** Index the leaves (the terminal nodes) of the tree by l_i , $i = 1, \dots, R$, where R is the number of leaves that will be identical to the number of the rules of the fuzzy classifier.
- **Step 1** Select a terminal node l_i ($i \in 1, \dots, R$), and collect all the attribute value tests $A_{i,j}$ related to the path of the selected terminal node to the root of the tree, where $j \in 1, \dots, n$.
- **Step 2** The $A_{i,j}$ attribute value tests (membership functions) define the antecedent part of the i th rule, and the conditional probabilities $P(c_1|ri), \dots, P(c_C|ri)$ at the

consequent part of the rule are given by the the example counts computed by the FID algorithm.

- **until** $i = R$ (all of the rules are recorded for all the leaves in the rule base system).

Post-pruning: Since FID is based on ID3 algorithm, it generates trees where every branches of the tree contain all of the fuzzy tests defined on the domain of the selected variable. Hence, the generated fuzzy decision tree is often more complex than is needed due to the addition of meaningless rules. This suggest that the simple transformation of a DT into a fuzzy model may be successfully followed by model-reduction steps to reduce the model's complexity and improve its interpretability.

The most straightforward way of the reduction of the rule base is that rules that are responsible for only a small number of data samples are erased from the rule-base, because they only cover exceptions or noise in the data. Hence, only the most important rules which determine the most the value of class labels are selected. The selection is based on the user defined $P(r_i) < \alpha_c$ (cut) parameter. The rules which have less activities than α_c are erased from the rule base.

4 Experimental Study

This section is intended to provide a comparative study based on a set of multivariate classification problems to present how the performance and the complexity of the FID based fuzzy classifier is varying with the several partitioning techniques. The chosen Iris, Wine, and Wisconsin data sets, coming from the UCI Repository of Machine Learning Databases (<http://www.ics.uci.edu>), are example of classification problems with different complexity. The decision trees were generated by using the following partitions: Ruspini-type partitioning (Ruspini) with triangular membership functions, the internal discretization method of the FID (FID), and fuzzy sets obtained by the proposed supervised fuzzy clustering algorithm (SC based). The results of these classifiers were compared to the results of the well known C4.5 algorithm. The effect of the previously presented post-pruning method has been also analyzed, (pruned-Ruspini) and (pruned-FID). During the experiments, performances of classifiers are measured by ten-fold cross validation. The performances of the classifiers generated by the C4.5, the FID, the pruned-FID and the pruned-SC based method are tested on the Iris, the Wisconsin and the Wine classification problems. The values in table 1 are resulted when the cut parameter is set to 0.0075. Besides the accuracy the complexity of the rule bases are also presented (in brackets) in the table. The underlined cases give the highest performance for a given classification problem. All the results for the three problems are demonstrative show that supervised clustering based method serve good classification performance. The other important analysis aspect could be the effect of the change of initial number of fuzzy sets. The performance of the previous classifiers are tested on the Wisconsin data while the number of initial fuzzy sets are changed between three and seven.

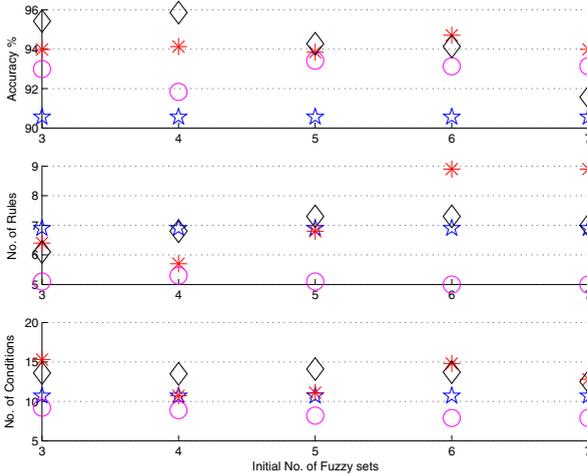


Fig. 4 Effects of the change of number of initial fuzzy sets, *notations*: *, \circ : pruned-FID, *, pruned-Ruspini, \diamond : pruned-SC based

The results of the runs are presented in fig. 4. The smallest classifiers are resulted in every cases when the pruning method is used for the FID method, while the accuracy exceeds the performance of the C4.5, but the supervised clustering based method is more accurate. Moreover the number of the rules in the proposed method is close to method C4.5, but the rules could be larger (see the third sub figure in fig. 4). It is interesting that the Ruspini-type partitioning serves close equal classification accuracies for all the initial number of fuzzy sets while the number of rules and conditions are unbalanced. The FID method without pruning gives rule bases with higher complexity.

Beside the detailed analysis of classifiers generated by several partitioning techniques, the proposed supervised clustering based classification method has been also compared to some other decision tree or rule-based classification models. The results for Wisconsin problem are summarized in table 2. As this analysis illustrates our pruning method could help to reduce the complexity of FID based decision trees.

Table 1 Best results of some methods ($\alpha_c = 0.0075$) for classification problems: av. accuracy (av. # of rules, av. # of conditions)

Problems	C4.5	FID	pr.-FID	pr.-SC based
Iris	93.33 (3.6,7.8)	95.33 (8.7,17.2)	94.67 (6.7,11.8)	96.00 (3.0,3.0)
Wine	94.90 (5.1,10.3)	93.73 (14.3,34.6)	93.73 (9.6, 20.2)	95.42 (21.6,68.1)
Wisc	92.71 (7.1,11.0)	94.71 (31.2,97.6)	93.57 (6.4,11.5)	95.86 (7.5,15.8)

Table 2 Comparison of some classifiers for the Wisconsin classification problem

Method	# Rules	# Conditions	Accuracy (%)
NeuroRule 1f	4	4	97.36
NeuroRule 2a	3	11	98.1
Full-RE	5	9	96.19
Fuzzy-GA1	1	4	97.07
Fuzzy-GA2	2	6	97.36
NEFCLASS	2	10-12	95.06
LID3-U	4	?	95.49
FIDMAT ($\alpha_c = 0.014$)	6.1	11.2	96.14
FIDMAT ($\alpha_c = 0.016$)	5.3	9.1	95.71

5 Conclusion

The generation of highly precise and interpretable models is the most frequent task of data mining. Also the interpretability of models and the nature of learning algorithms require effectively discretized (partitioned) features. Fuzzy partitioning of continuous features can increase the flexibility of the classifier since it has the ability to model fine knowledge details. Decision tree and rule induction methods often relay on *a priori* discretized (partitioned) continuous features. This paper demonstrates advantages of the application of supervised and fuzzy discretization of the attributes. A new algorithm based on supervised clustering has been proposed for the discretization of continuous features. It is shown that a proper rule structure is obtained by the proposed feature discretization, rule-induction and rule-optimization procedure.

Acknowledgements. The support from the TAMOP-4.2.2-08/1/2008-0018 (Livable environment and healthier people Bioinnovation and Green Technology research at the University of Pannonia, MK/2) project is gratefully acknowledged.

References

1. Abonyi, J., Szeifert, F.: Supervised fuzzy clustering for the identification of fuzzy classifiers. *Pattern Recognition Letters* 24(14), 2195–2207 (2003)
2. Breiman, L., Friedman, J., Olsen, R., Stone, C.: *Classification and Regression Trees*. The Wadsworth and Brooks/Cole Statistics/Probability Series. Wadsworth (1984)
3. Dougherty, J., Kohavi, R., Sahami, M.: Supervised and unsupervised discretization of continuous features. In: *International Conference on Machine Learning*, pp. 194–202 (1995)
4. Gallion, R., St. Clair, D., Sabharwal, C., Bond, W.: Dynamic id3: A symbolic learning algorithm for many-valued attribute domains. In: *Proceedings of the 1993 Symposium on Applied Computing*, pp. 14–20. ACM Press, New York (1993)
5. Janikow, C.: A genetic algorithm method for optimizing fuzzy decision trees. *Information Sciences* 89(3-4), 275–296 (1996)

6. Janikow, C.: Fuzzy decision trees: issues and methods. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 28, 1–14 (1998)
7. Michalski, R.: Learning flexible concepts: fundamental ideas and a method based on two-tiered representation in *Machine Learning III*
8. Pedrycz, W., Sosnowski, Z.: The design of decision trees in the framework of granular data and their application to software quality model. *Fuzzy Sets and Systems* 123(3), 271–290 (2001)
9. Quinlan, J.: Induction on decision trees. *Machine Learning* 1(1), 81–106 (1986)
10. Ruspini, E.: A new approach to clustering. *Information and Control* 15, 22–32 (1969)
11. Setnes, M.: *Complexity Reduction in Fuzzy Systems* (2001)

Fault Detection of the Tennessee Eastman Process Using Improved PCA and Neural Classifier

Mostafa Noruzi Nashalji, Mahdi Aliyari Shoorehdeli, and Mohammad Teshnehlab

Abstract. This paper describes hybrid multivariate method: Principal Component Analysis improved by Genetic Algorithm. This method determines main Principal Components can be used to detect fault during the operation of industrial process by neural classifier. This technique is applied to simulated data collected from the Tennessee Eastman chemical plant simulator which was designed to simulate a wide variety of faults occurring in a chemical plant based on a facility at Eastman chemical.

1 Introduction

Analysis of chemical data for detecting faults in chemical process has been intensively studied for the past decade [1-9], where a *fault* is an unpermitted deviation of at least one characteristic property or variable of the system from acceptable/ usual/ standard behavior. And also *fault detection* is a determination of fault present in the system and the time of detection, these definition suggested by the IFAC technical committee safe process.

Data driven technique has been applied in many chemical processes. Chiang et al. applied Fisher Discriminant Analysis (FDA), Discriminant Partial Least Squares (DPLS) and Principal Component Analysis (PCA) for fault detection and diagnosis of Tennessee Eastman Process (TEP) [10]. Russell et al. applied CVA and Dynamic PCA (DPCA) for fault detection in industrial processes [11]. Multi-Scale PCA (MSPCA) is used for fault detection and diagnosis in [12]. Chiang et al. is used FDA and Support Vector Machine (SVM) for fault diagnosis [13]. Chen & Liao is integrated two data driven techniques, neural network and PCA for fault monitoring [14].

Mostafa Noruzi Nashalji · Mahdi Aliyari Shoorehdeli · Mohammad Teshnehlab
Intelligent System Laboratory, Electrical Engineering Faculty,
K.N. Toosi University of Technology, Tehran, Iran
Research ID: C-7636-2009
e-mail: mostafanoruzi@hotmail.com

Zhang used improved kernel Independent Component Analysis (KICA) and SVM to the fault detection and diagnosis in the TEP [4]. Xiong et al. used PCA to reduce the dimension of the process data, present a new fault detection method based on artificial immune system for complicated process and illustrated proposed method by the TEP [6]. In [15], a SVM with knowledge incorporation is applied to detect the fault in TEP. Shao & Rong present nonlinear process monitoring based on maximum variance unfolding projections [16]. Xie & Kruger applied PSO-ICA and Support Vector Data Description (SVDD) for fault detection of TEP [17]. In [9] KICA, Kernel PCA (KPCA) and SVM are used for fault detection and diagnosis in the TEP.

The idea of key dimension selection in classification is selecting a subset of dimension in which classification result is better or comparable than the result obtained from the full set of dimensions. This paper is used PCA to produce a lower-dimensionality, PCA is a dimensionality reduction technique which transforms correlated original multivariate data to a set of uncorrelated data [18]. The proposed methodology is used GA to select Principal Component (PC) of data, which are collected of TEP. Selected PCs are used to make PCA matrix that is called transfer matrix. Neural classifier is used lower dimension reduced data to detect fault. The rest of this paper is organized as follows: TEP is present in section 2. In section 3 PCA introduced. Section 4 presents GA and the PC selection method based on GA. Section 5 discusses neural classifier. Section 6 shows result and simulation and conclusions coming in section 7.

2 Tennessee Eastman Process

The TEP is a well-known benchmark chemical process, which was firstly introduced by Downs and Vogel [19]. The TEP has been widely used by the process monitoring community as a source of data comparing various approaches [10, 11, 20-23]. The TEP was created by Eastman chemical company in open loop operation to provide a realistic industrial process for evaluating process control and monitoring methods [19].

The process consists of five major units: a reactor, condenser, compressor, separator and stripper. The TEP contain eight components: A, B, C, D, E, F, G and H and produces two products. TEP simulator coded in FORTRAN and a detailed description of the process and simulation is available [19]. The simulation code for the process in close loop and the data used for experiment are given by [24].

The TEP contains 41 measured and 12 manipulated variables from the controller, all measurements have Gaussian noise. The TEP simulation contains 21 pre-programmed faults. First simulation run was generated with no fault (fault 0) and each of other 20 simulation runs was generated under a different fault (fault 1-21). Sampling time was used to collect the simulated data for the training and testing data is 3 minutes. The simulations started with no faults and the faults were introduced one simulation hour into the run.

3 Principal Component Analysis

PCA is a dimensionality reduction technique [1]. PCA is used to linearly project a matrix of data in a low dimensional space, this space spanned by Principal Components (PCs) -i.e. eigenvectors corresponding to the large eigenvalues- for the distribution of the training data. PCA determines a set of orthogonal vectors, called loading vector, ordered by the amount of variance explained in the loading vector directions [1].

Let X is the original data set, where each row is a single sample of data set and each column is an observation. Data matrix $X \in R^{n \times m}$ consist n sample rows and m variable column that are normalized to zero mean and unit variance. The loading vectors are calculated by solving the stationary point of the optimization problem $\max_{V \neq 0} \frac{V^T X^T X V}{V^T V}$ where $V \in R^m$ is a loading vector [1]. PCA and Singular Value Decomposition (SVD) are related, the loading vector can be computed via the SVD, $\frac{1}{n-1}X = U \Sigma V^T$ (1) where $U \in R^{n \times n}$ and $V \in R^{m \times m}$ are unitary matrices, and the matrix $\Sigma \in R^{n \times m}$ contains the non-negative ($\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(n,m)} \geq 0$) and zero offdiagonal elements, the loading vectors are the orthonormal column vectors in the matrix V [1]. The variance associated with the i^{th} loading vector is equal to the square of the singular value σ_i .

A reduced set of a smaller number ‘a’ ($a < m$) of variables, is obtained by a set of loading vectors in the directions (PCs) where most of data variation occurs or good for fault detecting. The vector projected into the lower dimensional space, only ‘a’ variables needed to be monitored, as compared with m variables without the use of PCA. There are several methods for determining ‘a’ (reduction order’s value) [18, 23, 25, 26]. Some techniques for determining ‘a’ and selecting PCs that used in this paper are:

1. The largest singular value,
2. The PRESS statistic and
3. Genetic Algorithm

In the first technique, selecting the columns of the loading matrix $P \in R^{m \times a}$ to correspond to the loading vectors associated with the first ‘a’ singular values.

The dimension of the space can be determined using a cross validation procedure with the Prediction Residual Sum of Squares (PRESS) statistic [27],

$$PRESS(i) = \frac{1}{mn} \left\| X - \hat{X} \right\|_F^2$$

where i is the number of subset groups retained to calculate

\hat{X} and $\| \cdot \|_F$ is the Frobenius norm. The training set divided into groups. The PRESS statistic for one group is computed based on various dimensions of the space using all the other groups. This is repeated for each group, and the value i associated with the minimum average PRESS statistic determines the dimension of the space [1]. GA introduced in next section.

4 Genetic Algorithm

To make PCA matrix, the problem is, selecting a subset of PC. Finding out what PCs to use in classification task is called PC selection. PC selection is an optimization problem that involves searching the space of PC to find one subset that is optimal or near-optimal with respect to a certain criterion. A number of PC selection (some authors use the term “feature selection” rather than PC selection) methods have been proposed in [27-32]. GA provides a simple, general and powerful framework to select good subsets of PCs. GA is problem solving method that has been developed with evolution strategies.

Recently, GAs have attracted more and more attention in the feature -PC- selection area [32]. [32] and [33] presented one of the earliest studies of GA-based feature selection. In [31, 32, 34, 35] GA used for a feature selection. They used binary encoding and standard GA operators to solve the feature selection problem. GA is very efficient technique for PC selection in this paper, because the size of the selection area is enormous ($2^{52} - 1$).

The GA starts with randomly chosen parent chromosomes from the search space to create a population [36]. It works with chromosomes genotype. The population evolves towards the better chromosomes by applying genetic operators modeling the genetic processes occurring in the nature -selection, crossover and mutation- [36]. Sun et al. have employed a simple encoding scheme where the chromosome is a bit string whose length is determined by the number of eigenvectors [32]. Each eigenvector is associated with one bit in the string, if the i^{th} bit is ‘1’, then the i^{th} eigenvector is selected, otherwise, that component is ignored. Each chromosome represents a different subset of eigenvectors.

Initial population

The initial population is generated randomly.

Crossover

The bit string of two chromosomes, named parent, are cut into two pieces and the information of one side of the bit strings are swapped, cutting point chosen randomly [39]. Both offspring contains one part of information from one parent and the rest of the other parent.

Mutation

Mutation is the other evolutionary operator for GA; its general aim is to create a new individual from only one chromosome by changing one or more genes in it [38]. In this paper, mutation operator just flips a specific bit with a very low probability, it simply inverts on bit in the individual.

Selection

Crossover and mutation operators are made new chromosomes. Selection for replacement creates the new generation from the current one and the obtained offspring. The best N (N is equal initial population size) chromosomes in respect of fitness function selected the combined parent-offspring population.

Fitness evaluation

The fitness evaluation contains two terms: (1) accuracy and (2) the number of feature selected, PCs, [34].

The PC subset with fewer PCs and better performance is preferred. We used the fitness function shown below to combine the two terms: $fitness = miss + p \frac{N_f}{N_{at}}$ (2)

where $miss$ represents the misclassification rate of the data samples realized by the neural classifier, N_f is the number of PC chosen by the chromosome, N_{at} is the dimension of PC space and p is a parameter which balanced the misclassification rate and the number of retained feature.

5 Neural Classifier

A neural network consists of a large number of simple processors called neurons. The input vector –in this paper called neural input- is containing r -inputs $X = [x_1, x_2, \dots, x_r]$. These inputs are from external source or can come from other units. Neural inputs multiplied by the weight matrix W to form WX . Each neuron has a constant input of ‘1’ shown by x_0 , x_0 is multiplied by a bias (w_0) to form w_0x_0 . In each neuron WX and w_0x_0 send to the summer. The summer output, n , is $WX + w_0x_0$ ($n_i = \sum_{q=0}^r w_{iq}x_q$ $i = 1, \dots, s$ neurons of single layer perceptron). The output of neuron i is $a_i = f(n_i)$, where $f(\cdot)$ is an activation function of the neurons [39, 40]. The output of the single layer perceptron is given as $a = [a_1, a_2, \dots, a_p]^T$.

Network with several layers has simply cascade three perceptron networks, each layer has its own weight matrix W , its bias vector w_0 and as output vector a . different layers can have different numbers of neurons. A layer whose output is the network output is called an output layer; the other layers are called hidden layers [40]. The network has an output layer (layer 3) and two hidden layers (layer 1 and 2). Each neuron in the first hidden layer has an input from every neural input with one additional input of 1 -bias-; each neuron in the second hidden layer has an input from every neuron in the first layer additional input like first layer. In the output layer, there is single neuron called output neuron. The output neuron has an input from every neuron in the second hidden layer and one additional input like first and second layer. In fault detection application, neural networks used measured and manipulated variables as inputs, while the output represents categories – this neural network called neural classifier- in this paper contains faulty and normal operation. Usually the output is dummy variable (‘1’ or ‘0’) where ‘1’ indicates an in-class member while ‘0’ indicates a non-class member. Here, dummy variables as output are ‘1’ or ‘-1’ which the margin between the in-class members –shown normal operation- and non-class members -shown faulty operation- is bigger than previous dummy variable.

6 Result and Discussion

Methods used in this paper are:

- Fault detection using all variables by neural classifier.
- Fault detection using PCA based on largest singular value with first 15 columns of the loading vector in the matrix V at equation (1) by neural classifier.
- Fault detection using PCA based on PRESS statistic by neural classifier: 400 subset groups of PCs are generated randomly to find good PCs subset by PRESS statistic for fault detecting by neural classifier.
- Fault detection using improved PCA, PCA that used GA to select PCs, by neural classifier: GA has several parameters for which no guidance is available on how to specify their values. We used a population size of 100 and 100 generations, with a probability of mutation $p_m = 0.02$ and probability of crossover $p_c = 0.3$ to find near-optimal PC subset. We used $p = 30$ which is a balanced parameter in fitness function of GA -equation (2)-.

Neural classifier has three layers, first hidden layer has 25 neurons, second hidden layer has 10 neurons and output layer has one neuron. The activation function of each neuron of hidden layers is hyperbolic tangent sigmoid and output's activation function is linear. Neural classifier is trained using Levenberg-Marquardt learning algorithm [41]. The overall missed detection rate for each method when applied to all disturbances of the testing set and selected PCs are listed in Table 1. The sensitivity of fault detection techniques was quantified by calculating the missed detection rate for fault 1-21 of the testing set, for each method. The missed detection rate for all 21 faults as faulty state were computed and tabulated in Table 2. The detection delay for all 21 faults is listed in Table 3. The detection delay is recorded as the first time instant in which method exceeded. The numbers in table 1-3 are average of 30 runs of each method.

Classification results using all variables

For the TEP data, neural classifier achieved bad detection result by using all variable. This method produced higher missed detection rate than PCA based methods.

Table 1 Overall missed detection rate and order for the various methods

Method	Overall missed detection rate	order
all variable	93.04	52
PCA based largest singular value	93.25	15
PCA based PRESS statistic	94.65	15
Improved PCA	37.1	15

Table 2 Missed detection rate for the testing set

fault	all variable	PCA based largest singular value	PCA based PRESS statistic	Improved PCA
1	94.29	96.54	92.79	16.73
2	86.36	78.45	88.80	15.99
3	94.67	96.10	96.72	53.44
4	95.77	94.99	96.52	28.63
5	94.12	95.28	96.15	57.94
6	91.9	92.46	88.30	17.87
7	94.10	89.79	93.67	13.37
8	93.65	94.9	95.49	28.37
9	95.02	94.93	96.49	51.04
10	94.62	96.54	96.72	53.46
11	95.74	95.5	96.21	44.62
12	94.94	96	95.16	33.4
13	89.38	89.97	92.11	18.60
14	94.83	91.72	94.35	25.53
15	93.15	94.71	95.76	61.99
16	95.04	96.18	96.65	48.52
17	91.83	96	97.07	47.92
18	88.99	85.29	90.75	27.5
19	91.98	92.36	95.12	57.07
20	95.22	95.75	96.42	25.72
21	88.36	94.67	96.43	51.36

Classification results with Principal Component selection

The PCA based on largest singular value and PRESS statistic performed better than the fault detection using all variable but these methods have high missed detection rate.

It is difficult to compare the GA method with the other methods. Unlike the other methods, this method does not try to find the best subset of a specified size its search space surrounds all the subsets.

The improved PCA with MLP classifier obtained a good accuracy using a few of PCs.

7 Conclusions

In fault detection methods, the number of variable or PC greatly affect the ability of fault detection. This paper have focused on performing PC selection on data sets since PC selection is typically done in an off-line mode, the execution time of a particular algorithm is of much less importance than its ultimate classification performance.

This paper illustrates the deservingness of various methods of PC selection to detect fault of TEP by neural classifier. It was shown that the optimum or near optimum number of PCs, which maximizes the sensitivity of fault detection by neural classifier, can be select by GA. Results show that the proposed method

Table 3 Detection delay (minutes) for the testing set

fault	all variable	PCA based largest singular value	PCA based PRESS statistic	Improved PCA
1	17.1	19.55	15.1	61.99
2	29.6	33.7	6.4	48.52
3	10.7	19.35	24.5	47.92
4	17.9	11.38	10.4	27.5
5	23.6	29	10.4	57.07
6	18.5	20.17	8.9	25.72
7	20.2	37.7	10	51.36
8	20.8	32	19.4	61.99
9	77.8	56.59	1.6	48.52
10	25.2	41.8	4.9	47.92
11	8.9	11.17	11.1	27.5
12	18.9	37.2	6.9	57.07
13	26.1	45.4	5.3	25.72
14	14	4.655	26	51.36
15	2	22.24	3.4	61.99
16	14.2	84.8	14.2	48.52
17	19.1	35.79	78.2	47.92
18	10.5	42.6	8.2	27.5
19	26.8	31.76	8.9	57.07
20	26.7	70.2	52.6	25.72
21	15.2	81.6	10.6	51.36

performs good detection capability. While the improved PCA with neural classifier had the lowest missed detection rate and the lowest detection delay of most faults for the TEP, it should always be used as part of a fault detection strategy.

References

- [1] Chiang, L.H., Russell, E.L., Braatz, R.D.: *Fault Detection and Diagnosis in Industrial Systems*. Springer, New York (2001)
- [2] Kourtji, T.: Process analysis and abnormal situation detection: from theory to practice. *IEEE Control Systems Magazine* 22(5), 10–25 (2002)
- [3] MacGregor, J.F., Jaeckle, C., Kiparissides, C., Koutoudi, M.: Process monitoring and diagnosis by multiblock PLS method. *American Institute of Chemical Engineering Journal* 40(5), 826–838 (1994)
- [4] Zhang, Y.: Fault Detection and Diagnosis of Nonlinear Processes Using Improved Kernel Independent Component Analysis (KICA) and Support Vector Machine (SVM). *Industrial & Engineering Chemistry Research* 47(18), 6961–6971 (2008)

- [5] Wang, L., Yu, J.: A Modified Discrete Binary Ant Colony Optimization and Its Application in Chemical Process Fault Diagnosis. In: Wang, T.-D., et al. (eds.) SEAL 2006. LNCS, vol. 4247, pp. 889–896. Springer, Heidelberg (2006)
- [6] Xiong, C., Zhao, Y., Liu, W.: Fault Detection Method Based on Artificial Immune System for Complicated Process. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) ICIC 2006. LNCS (LNAD), vol. 4114, pp. 625–630. Springer, Heidelberg (2006)
- [7] Musulin, E., Yélamos, I., Puigjaner, L.: Integration of principal component analysis and fuzzy logic systems for comprehensive process fault detection and diagnosis. *Industrial and Engineering Chemistry Research* 45(5), 1739–1750 (2006)
- [8] Ge, Z., Song, Z.: Process Monitoring Based on Independent Component Analysis–Principal Component Analysis (ICA–PCA) and Similarity Factors. *Industrial and Engineering Chemistry Research* 46(7), 2054–2063 (2007)
- [9] Zhang, Y.: Enhanced statistical analysis of nonlinear processes using KPCA, KICA and SVM. *Chemical Engineering Science* 64(5), 801–811 (2009)
- [10] Chiang, L.H., Russell, E.L., Braatz, R.D.: Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis. *Chemometrics and Intelligent Laboratory Systems* 20(2), 243–252 (2000)
- [11] Russell, E.L., Chiang, L.H., Braatz, R.D.: Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis. *Chemometrics and Intelligent Laboratory Systems* 51(1), 81–93 (2000)
- [12] Misra, M., Yue, H.H., Qin, S.J., Ling, C.: Multivariate process monitoring and fault diagnosis by multi-scale PCA. *Computers and Chemical Engineering* 26(9), 1281–1293 (2002)
- [13] Chiang, L.H., Kotanchek, M.E., Kordon, A.K.: Fault diagnosis based on Fisher discriminant analysis and support vector machines. *Computers & Chemical Engineering* 28(8), 1389–1401 (2004)
- [14] Chen, J., Liao, C.-M.: Dynamic process fault monitoring based on neural network and PCA. *Journal of Process Control* 12(2), 277–289 (2002)
- [15] Akbaryan, F., Bishno, P.R.: Fault diagnosis of multivariate systems using pattern recognition and multisensor data analysis technique. *Computers & Chemical Engineering* 25(9-10), 1313–1339 (2001)
- [16] Shao, J.-D., Rong, G.: Nonlinear process monitoring based on maximum variance unfolding projections. *Expert Systems with Applications* 36(8), 11332–11340 (2009)
- [17] Xie, L., Kruger, U.: Statistical processes monitoring based on improved ICA and SVDD. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) ICIC 2006. LNCS, vol. 4113, pp. 1247–1256. Springer, Heidelberg (2006)
- [18] Jackson, J.E.: A users guide to principal components analysis. Wiley, New York (1991)
- [19] Downs, J.J., Vogel, E.F.: A plant-wide industrial process control problem. *Computers & Chemical Engineering* 17(3), 245–255 (1993)
- [20] Raich, A., Çinar, A.: Statistical process monitoring and disturbance diagnosis in multivariable continuous processes. *American Institute of Chemical Engineers (AIChE)* 42(4), 995–1009 (2004)
- [21] Ku, W., Storer, R.H., Georgakis, C.: Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and Intelligent Laboratory Systems* 30(1), 179–196 (1995)
- [22] Gertler, J., Li, W., Huang, Y., McAvoy, T.: Isolation enhanced principal component analysis. *American Institute of Chemical Engineers (AIChE)* 45(2), 323–334 (2004)
- [23] Himes, D.M., Storer, R.H., Georagkis, C.: Determination of the Number of Principal Components for Disturbance Detection and Isolation. In: Proceedings of the American control conference, June 1994, pp. 1279–1283 (1994)

- [24] Multiscale Systems Research Laboratory, <http://brahms.scs.uiuc.edu>
- [25] Qin, S.J., Dunia, R.: Determining the number of principal components for best reconstruction. *Journal of Process Control* 10(2), 245–250 (2000)
- [26] Valle, S., Li, W., Qin, S.J.: Selection of the Number of Principal Components: The Variance of the Reconstruction Error Criterion with a Comparison to Other Methods. *Industrial and Engineering Chemistry Research* 38(11), 4389–4401 (1999)
- [27] Wold, S.: Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics* 20, 397–405 (1978)
- [28] Dash, M., Liu, H.: Feature selection for classification. *Intelligent Data Analysis* 1(3), 131–156 (1997)
- [29] Blum, A.L., Langley, P.: Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence* 97, 245–271 (1997)
- [30] Sun, Z., Bebis, G., Miller, R.: Boosting Object Detection Using Feature Selection. In: *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS 2003)*, pp. 290–296 (2003)
- [31] Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(1), 4–37 (2000)
- [32] Yang, J., Honavar, V.: Feature subset selection using a genetic algorithm. *IEEE Intelligent Systems and their Applications* 13(2), 44–49 (1998)
- [33] Sun, Z., Bebis, G., Miller, R.: Object detection using feature subset selection. *Pattern Recognition* 37(11), 2165–2176 (2004)
- [34] Siedlecki, W., Sklansky, J.: A note on genetic algorithm for large-scale feature selection. *Pattern Recognition Letters* 10, 335–347 (1989)
- [35] Chtioui, Y., Bertrand, D., Barba, D.: Feature selection by a genetic algorithm. Application to seed discrimination by artificial vision. *Journal of the science of food and agriculture* 76, 77–86 (1998)
- [36] Choa, H.-W., et al.: Genetic algorithm-based feature selection in high-resolution NMR spectra. *Expert Systems with Applications: An International Journal* 35(3), 967–975 (2008)
- [37] Shopova, E.G., Vaklieva-Bancheva, N.G.: BASIC—A genetic algorithm for engineering problems solution. *Computers and Chemical Engineering* 30(8), 1293–1309 (2006)
- [38] Nelles, O.: *Nonlinear system identification*. Springer, Berlin (2001)
- [39] Gupta, M.M., Jin, L., Homma, N.: *Static and dynamic neural networks: from fundamentals to advanced theory*. Wiley-IEEE (2003)
- [40] Hagan, M.T., Demuth, H.B., Beale, M.H.: *Neural Network Design*. PWS Publishing, Boston (1996)
- [41] Hagan, M.T., Menhaj, M.: Training feed-forward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks* 5(6), 989–993 (1999)

Part II
New Evolutionary Methodologies

Bare Bones Particle Swarm Applied to Parameter Estimation of Mixed Weibull Distribution

Renato A. Krohling, Mauro Campos, and Patrick Borges

Abstract. An approach for estimating the parameters of mixed Weibull distributions is presented. The problem is formulated as maximization of the likelihood function of the corresponding mixture model. For the solution of the optimization problem, Bare Bones Particle Swarm Optimization (BBPSO) algorithm is applied. Illustrative example for a case study using censored data are provided in order to show the suitability of the BBPSO algorithm for this kind of problem very common in lifetime modelling.

1 Introduction

The Weibull distribution has commonly been used for modeling of lifetimes in reliability. However when the lifetimes have more than one failure cause, then the mixed Weibull distribution is more appropriate [1]. The mixed Weibull distribution is used in applications in different areas, e.g., reliability modeling [2], in analysis of automotive reliability [3], in the distribution of time between failures of machining center [4], for estimation of wind speed distributions [5].

In this paper, an approach for estimating the parameters of mixed Weibull distributions is presented. Different approaches have been developed to tackle this

Renato A. Krohling

Departamento de Informática, PPGI. Universidade Federal do Espírito Santo. Av. Fernando Ferrari 514, Goiabeiras, 29075-910, Vitória ES, Brazil
e-mail: krohling.renato@gmail.com

Mauro Campos · Patrick Borges

Departamento de Estatística, CCE. Universidade Federal do Espírito Santo. Av. Fernando Ferrari 514, Goiabeiras, 29075-910, Vitória ES, Brazil
e-mail: maurocmc@cce.ufes.br, patrick@cce.ufes.br

problem. A graphical approach was proposed by Jiang and Murthy [6] which provides quite crude parameters estimation. Kececioglu and Wendai [1] introduced an algorithm which combines the Least-Squares method with the Bayes' Theorem. Jiang and Kececioglu [7] presented an algorithm which follows the principle of the maximum likelihood estimation (MLE) through the EM (Expectation and Maximization) algorithm. Bayesian methods for estimating the parameters and the reliability function have been developed for censored data [8, 9], for statistical analysis of gate oxide breakdown mechanisms [10].

MLE of the parameters of a mixed Weibull distribution based on censored data consists of a difficult optimization problem. Gradient methods may get trapped into local optima and fail to converge to global solutions. In the last few years, alternative optimization methods like genetic algorithms [11], simulated annealing [12], and a shuffled complex-evolution metropolis algorithm named SCEMA-UA [13] have also been proposed to estimate the parameters of a unique Weibull distribution with promising results.

Particle swarm optimization (PSO) developed by Kennedy and Eberhart [14] is an algorithm which belongs to the Swarm Intelligence methods [15] which has shown to be a very powerful optimization method used in different areas of science and engineering. It is an optimization method which requires only the evaluation of the objective function (zero-order method). In this paper, the PSO-algorithm is used for MLE of the parameters of a mixed Weibull distribution based on non-censored and censored data. As far as we know, for the first time introduced in this context. The rest of the paper is organized as follows: in section 2, the problem is described. The Bare Bones PSO are explained in section 3. Section 4 provides simulation results and discussions, followed by conclusions in section 5.

2 Problem Formulation

Let T be a nonnegative random variable representing the lifetimes of elements in a population. Suppose that T is a continuous random variable with probability density function $f(\cdot; \theta)$ where $\theta = (\theta_1, \dots, \theta_k)$ is a vector of parameters taking on values in a set Θ . Suppose further that t is a nonnegative real number. The probability that an element of the population survives longer than t is given by the reliability function

$$S(t) = S(t; \theta) = \Pr(T > t) = \int_t^{\infty} f(u; \theta) du \quad (1)$$

where $\Pr(\cdot)$ stands for the probability of the event in parentheses. The reliability function is a monotone decreasing continuous function with $S(0) = 1$ and $S(t) \rightarrow 0$ as $t \rightarrow \infty$.

In statistical inference we use a sample of data to draw inferences about some aspect of the distribution from which the data were taken. Often the inference concerns the value of one or more unknown parameters, which describe some attribute of the distribution. However, lifetime data come with a feature that creates special difficulties in the analysis of the data. This feature is known as censoring and, broadly

speaking, occurs when exact lifetimes are known for only a portion of elements under study. Formally an observation is said to be right censored at C if the exact value of the observation is not known but only that it is greater than or equal to C . Similarly, an observation is said to be left censored at C if it is known only that the observation is less than or equal C (for more details see Lawless [16]). In this work, only right censoring will be discussed, though many of ideas transfer in an obvious way to the case of left censoring. Hereafter the term censoring will be used meaning in all situations right censoring.

Now we assume that the notation $(t, \delta) = \{(t_i, \delta_i); i = 1, \dots, n\}$ represents a set of n observations from the distribution of T , where $\delta_i = 0$ if t_i is a censored observation, and $\delta_i = 1$ otherwise. Then the likelihood function is given by

$$L(\theta) = L(\theta; t) \propto \prod_{i=1}^n [f(t_i; \theta)]^{\delta_i} [S(t_i; \theta)]^{1-\delta_i} \tag{2}$$

and the log-likelihood function is defined by $l(\theta) = \log L(\theta)$.

Discrete mixture models arise when individuals in a population are each one of m distinct types, with a proportion p_j of the subpopulation being of the j -th type; the ω_j 's satisfy $0 < \omega_j < 1$ and $\sum_j \omega_j = 1$. Individuals of type j are assumed to have a lifetime distribution with reliability function $S_j(t)$. An individual randomly selected from this population has its reliability function given by

$$S(t) = \omega_1 S_1(t) + \dots + \omega_m S_m(t). \tag{3}$$

Models of this kind are termed discrete mixture models and are useful in situations where the population is non-homogeneous but it is not possible to distinguish between individuals of different types. Often the $S_j(t)$'s in Eq. (3) are taken from the same parametric family, though this is, of course, unnecessary. The properties of a mixture model are easily derived from the properties of the m distributions, or components, involved in the mixture. Models for m larger than 2 or 3 are rarely used, unless one is in a situation where the $S_j(t)$'s are completely known. Otherwise, the number of unknown parameters is usually large, and estimating them is difficult enough to render the models unattractive. Even models with $m = 2$ or 3 are difficult to handle statistically.

Let us consider a population consisting of a mixture of two independent sub-populations and each sub-population with its own unique failure mode and distribution. So, the lifetime distribution of the mixed population is given by

$$f(t) = \omega_1 f_1(t) + \omega_2 f_2(t), \tag{4}$$

where $f(t)$ is the probability density function (pdf) of the mixed population, and $f_j(t)$ is the pdf of the j -th subpopulation with $j = 1, 2$. The parameters ω_1 and ω_2 are weights, where $\omega_1, \omega_2 \in (0, 1)$ with $\omega_2 = 1 - \omega_1$. For the Weibull distribution, the pdf of each subpopulation $f_j(t)$ is described by

$$f_j(t) = (\beta_j / \alpha_j^{\beta_j}) \cdot t^{\beta_j - 1} \cdot \exp \left[- (t / \alpha_j)^{\beta_j} \right] \tag{5}$$

where β_j and α_j are the shape and scale parameters of each distribution, respectively. Therefore, for a mixture of two Weibull distributions, five parameters need to be estimated and the parameter vector θ to be estimated is made up of $\theta = [\beta_1, \alpha_1, \beta_2, \alpha_2, \omega_1]$. So, the optimization problem is formulated as

$$\theta^* = \arg \max L(\theta) = \arg \max \left\{ \prod_{i=1}^n [f(t_i; \theta)]^{\delta_i} [S(t_i; \theta)]^{1-\delta_i} \right\}. \quad (6)$$

In this work, PSO is applied to solving the maximum likelihood estimation of the parameters of a mixture of two-Weibull distribution formulated as an optimization problem.

3 Bare Bones Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a population-based algorithm [15]. PSO is initialized with a population of candidate solutions. Each candidate solution in PSO, called particle, has associated a randomized velocity. Each particle moves through the search space and keeps track of its coordinates in the search space, which is associated with the best solution (fitness) it has achieved so far, $pbest$. Another $best$ value tracked by the global version of the particle swarm optimizer is the overall best value, $gbest$, and its location, obtained so far by any particle in the population.

In this work we focus on the Bare Bones PSO [17]. In the Bare Bones PSO (BBPSO) two Gaussian random number generators are used to sampling the search space based on the global best ($gbest$) and the personal best ($pbest$) particle as follows:

$$\theta_{I,J} \stackrel{d}{=} N(\mu_{I,J}, \sigma_{I,J}^2) \quad (7)$$

where N designates the Gaussian distribution with mean $\mu_{I,J} = (gbest_J + pbest_{I,J})/2$ and standard deviation $\sigma_{I,J} = |gbest_J - pbest_{I,J}|$ for each variable $J = 1, \dots, k$ of the particle I . This version of PSO presents some advantages over other versions because the designer needs just to specify the swarm size (population size). The BBPSO algorithm is described as follows:

BBPSO algorithm

- **Input parameters:** Swarm size
- **For** each particle I
- // random initialization of a population of particles with positions θ_I using uniform probability distribution, where $\underline{\theta}_I$ and $\bar{\theta}_I$ are the lower and upper bound respectively
- $\theta_I = \underline{\theta}_I + (\bar{\theta}_I - \underline{\theta}_I)U_I(0, 1)$
- $pbest_I = \theta_I$
- Compute $L(\theta_I)$ // calculation of the likelihood

- $p_{gbest} = \arg \max L(\theta_I)$ // global best particle
- **End for**
- **Do**
- **For** each particle I
- Update position according to (7)
- Compute $L(\theta_I)$ // calculation of the likelihood
 - **If** $L(\theta_I) > L(p_{best_I})$ **then** $p_{best_I} = \theta_I$ // update of the personal best
 - **If** $L(\theta_I) > L(p_{gbest})$ **then** $p_{gbest} = \theta_I$ // update of the global best
- **End for**
- **While** termination condition not met.
- **Output:** $p_{gbest} = \theta^* = [\beta_1, \alpha_1, \beta_2, \alpha_2, \omega_1]$.

The termination criterion of the BBPSO algorithm is usually the maximum number of iterations, which is pre-specified by the designer and depends on the problem being solved. In a previous attempt developed by Campos, Krohling and Borges [18] PSO was applied for estimating the parameters of the generalized Gamma family with very promising results. In the following, we apply the BBPSO algorithm to illustrate the approach.

4 Results

Next, we present a case study for estimating the parameters of mixed Weibull using a real data set.

Table 1 Data set for the case study

478	484+	583	626+	753	753	801	834	850	944
959	1071+	1318	1377	1472+	1534	1579+	1610+	1729+	1792+
1847+	2400	2550+	2568+	2639	2944	2981	3392	3392	3791+
3904	4443+	4829	5328	5562	5900+	6122	6226+	6331	6531
6711+	6835+	6947+	7878+	7884+	10263+	11019+	12986	13103+	23245+

Case study. The data are actual failures of the throttle for 25 pre-production prototype general-purpose load-carrying vehicles. The data comprise both failure and right-censored observations. The data set given in Table 1 is taken from Jiang and Murthy [6]. Observations with the symbol “+” represent censored distances.

The estimated parameters are as follows: $\beta_1 = 1.03$, $\alpha_1 = 8702.02$, $\beta_2 = 4.21$, $\alpha_2 = 16831.50$, $\omega_1 = 1.0$ and $\omega_2 = 0.0$. Fig. 1 depicts the convergence of the BBPSO algorithm, which occurs in the first 50 iterations. In order to test the estimated parameters for the Example studied, we carried out the Kaplan-Meier non-parametric estimation method. For more details on this estimation method, see

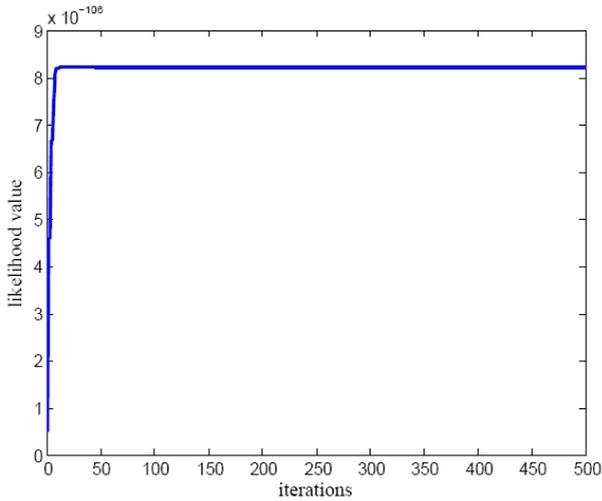


Fig. 1 Convergence of BBPSO algorithm for the case study

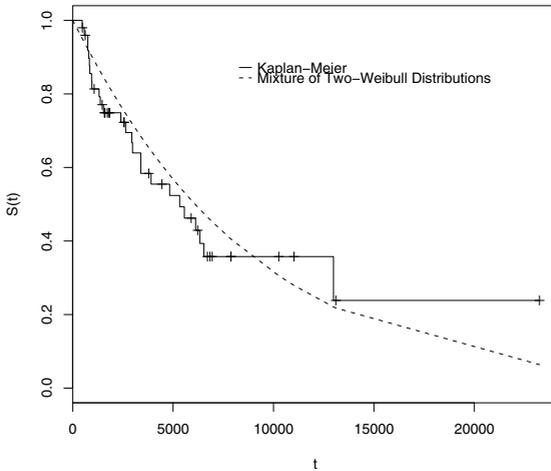


Fig. 2 The step function is the Kaplan-Meier estimate of the reliability function for the case study. The smooth curve is the fitted reliability function for the case study using BBPSO.

Appendix. Fig. 2 shows the Kaplan-Meier (KM) estimates of the reliability function for this Example, and the estimated reliability function using BBPSO. As we can observe in Fig. 2 the estimated reliability function and the KM non-parametric estimate is not that good, especially for large values of t . It is worth to mention, that in this case, 24 out of 50 are censored data, which means that the information contained in these data are uncertain showing limitations of the approach.

5 Conclusion

In this paper, estimating the parameters of a mixed Weibull distribution has been formulated as the maximum likelihood estimation (MLE) problem. For the solution of the maximization problem a Bare Bones Particle Swarm Optimization (BBPSO) algorithm has been proposed. BBPSO presents interesting features as optimization algorithm because its convergence, effectiveness, robustness, and simplicity of implementation. These features have attracted its attention to solve nonlinear, non-differentiable, multimodal optimization problems. We apply BBPSO to find out the parameters of mixed Weibull distributions by solving the maximum likelihood estimation problem. The simulations carried out for a case study with censored data demonstrated that BBPSO finds the parameters of the Weibull distributions (shape and scale) as well as the weights.

To the best of our knowledge, the approach presented here using BBPSO is the first attempt to tackle this kind of problem. Work in progress is investigating the application of more powerful versions of PSO [19] for challenging optimization problems in the field of Statistics and Reliability Engineering, where solutions in closed form do not exist yet.

Acknowledgements. R.A. Krohling thanks the financial support of the FAPES/MCT/CNPq (DCR grant 37286374/2007) for funding this research project in form of a fellowship.

References

1. Kececioglu, D.B., Wendai, W.: Parameter estimation for mixed-Weibull distribution. In: Proc. of the Annual Symposium on Reliability and Maintainability, pp. 247–252 (1998)
2. Patra, K., Dey, D.K.: A multivariate mixture of Weibull distributions in reliability modeling. *Statistics and Probability Letters* 45, 225–235 (1999)
3. Majeske, K.D.: A mixture model for automobile warranty data. *Reliability Engineering and System Safety* 81, 71–77 (2003)
4. Dai, Y., Zhou, Y.-F., Jia, Y.-Z.: Distribution of time between failures of machining center based on type I censored data. *Reliability Engineering and System Safety* 79, 377–379 (2003)
5. Carta, J.A., Ramírez, P.: Analysis of two-component mixture Weibull statistics for estimation of wind speed distributions. *Renewable Energy* 32, 518–531 (2007)
6. Jiang, R., Murthy, D.N.P.: Modeling failure-data by mixture of 2 Weibull distributions: a graphical approach. *IEEE Transactions on Reliability* 44, 477–488 (1995)
7. Jiang, S., Kececioglu, D.: Maximum likelihood estimates, from censored data, for mixed-Weibull distributions. *IEEE Transactions on Reliability* 41, 248–255 (1992)
8. Sinha, S.K.: Bayesian estimation of the parameters and reliability function of a mixture of Weibull life distributions. *Journal of Statistical Planning and Inference* 16, 377–387 (1987)
9. Chen, K.-W., Papadopoulos, A.S., Tamer, P.: On Bayes estimation for mixtures of two Weibull distributions under type I censoring. *Microelectronics and Reliability* 29, 609–617 (1989)
10. Tan, C.M., Raghavan, N.: An approach to statistical analysis of gate oxide breakdown mechanisms. *Microelectronics Reliability* 47, 1336–1342 (2007)

11. Thomas, G.M., Gerth, R., Velasco, T., Rabelo, L.C.: Using real-coded genetic algorithms for Weibull parameter estimation. *Computers and Industrial Engineering* 29, 377–381 (1995)
12. Abbasi, B., Jahromi, A.H.E., Arkat, J., Hosseinkouchack, M.: Estimating the parameters of Weibull distribution using simulated annealing algorithm. *Applied Mathematics and Computation* 183, 85–93 (2006)
13. Gong, Z.: Estimation of mixed Weibull distribution parameters using the SCEM-UA algorithm: Application and comparison with MLE in automotive reliability analysis. *Reliability Engineering and System Safety* 91, 915–922 (2006)
14. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: *Proc. of the IEEE International Conference on Neural Networks*, pp. 1941–1948 (1995)
15. Kennedy, J., Eberhart, R., Shi, Y.: *Swarm intelligence*. Morgan Kaufmann Publishers, San Francisco (2001)
16. Lawless, J.F.: *Statistical models and methods for lifetime data*. John Wiley and Sons, New York (1982)
17. Kennedy, J.: Bare bones particle swarms. In: *Proc. of the IEEE Swarm Intelligence Symposium SIS 2003*, pp. 80–87 (2003)
18. Campos, M., Krohling, R.A., Borges, P.: Particle swarm optimization for inference procedures in the generalized gamma family based on censored data. In: Mehnen, J., et al. (eds.) *Applications of Soft Computing: From Theory to Praxis*. Springer, Heidelberg (2009)
19. Krohling, R.A., Mendel, E.: Bare bones particle swarm with Gaussian and Cauchy jumps. In: *Proc. of the IEEE Congress on Evolutionary Computation CEC 2009*, pp. 3285–3291 (2009)

Appendix

The Kaplan-Meier Estimates of the Reliability Function

The Kaplan-Meier (KM) estimator is a non-parametric estimator for the reliability function [16]. By drawing a graph of the reliability function estimated and overlapping with the KM estimate, we are only comparing the parametric estimate with the non-parametric estimate of the reliability function. When the two curves overlap, it is a graphics indication that the probabilistic model has been good fitted, i.e., it is a good sign that the parameters of the model has been well estimated.

Let n be the total number of elements whose survival times, censored or not, are available. Relabel the n survival times in order of increasing magnitude such that $t_{(1)} \leq t_{(2)} \leq \dots \leq t_{(n)}$. The Kaplan-Meier estimator of the reliability function $S(t)$ is given by

$$\hat{S}(t) = \prod_r \frac{n-r}{n-r+1} \quad (8)$$

where r runs through those positive integers for which $t_{(r)} \leq t$ and $t_{(r)}$ is uncensored. The variance of the Kaplan-Meier estimator is approximated by

$$V(\hat{S}(t)) \approx \sum_r \frac{1}{(n-r)(n-r+1)}. \quad (9)$$

Structural Optimization Using Harmony Search Algorithm

D. Srikanth and S.V. Barai

Abstract. The paper presents the Structural optimization based on the New Harmony Search (HS) meta-heuristic algorithm. HS was conceptualized using the musical process of searching for a perfect state of harmony. The HS algorithm does not require initial values and uses a random search instead of a gradient search. 3D truss structure examples with fixed geometries are presented to demonstrate the effectiveness and robustness of the new method. The results indicate that the new technique is an efficient search and optimization method for solving complex structural engineering problems in comparison to conventional approaches.

1 Introduction

Engineering Design has become a major challenge in today's increasingly complex world. The rigorous exercise of arriving at a best or optimum design under a predefined design window is termed as the process of optimization. Structural optimization is the process of obtaining maximum or minimum value of an objective function while subjected to various constraints. Thus the motive of optimal design is to obtain best possible design following a set of pre-selected measures of effectiveness. The awareness of the scarcity of natural resources drives us towards lightweight and low cost structures, thus emphasizing the need for weight and cost optimization of structures. Designers prefer to minimize the volume or the weight of the structure by optimization. Many traditionally mathematical optimization algorithms have been used in structural optimization problems [1-3]. However, most of these algorithms are limited for the structure design. Recently, evolutionary algorithms (EAs) such as genetic algorithms, evolutionary programming and evolution strategies have been attractive because they do not apply mathematical assumptions to the optimization problems and have better global search abilities over conventional optimization algorithms [4-7].

The computational drawbacks of mathematical methods (i.e., complex derivatives, sensitivity to initial values, and the large amount of enumeration memory required) have forced researchers to rely on meta-heuristic algorithms based on

D. Srikanth · S.V. Barai
Department of Civil Engineering, Indian Institute of Technology, Kharagpur, India

simulations to solve optimization problems. The common factor in meta-heuristic algorithms is that they combine rules and randomness to imitate natural phenomena. These phenomena include the biological evolutionary process. In the last decade, these meta-heuristic algorithms, especially the genetic algorithm have been broadly applied to solve various structural optimization problems, and have occasionally overcome several deficiencies of conventional mathematical methods. To solve complicated optimization problems, however, new heuristic and more powerful algorithms based on analogies with natural or artificial phenomena remain to be explored. Geem's research group has implemented a new Heuristic Optimization Algorithm: Harmony Search [8-19]. The New Harmony search meta-heuristic algorithm was conceptualized using the musical process of searching for a perfect state of harmony. Compared to mathematical optimization algorithms, the HS algorithm imposes fewer mathematical requirements and does not require initial values for the decision variables. Furthermore, the HS algorithm uses a random search, which is based on the harmony memory considering rate and the pitch adjusting rate (these are defined in the following section), instead of a gradient search, so derivative information is unnecessary. Although the HS algorithm is a comparatively simple method, it has been successfully applied to various optimization problems including the traveling salesperson problem, the layout of pipe networks, pipe capacity design in water supply networks; hydrologic model parameter calibrations; cofferdam drainage pipe design and optimal school bus routings [8-23].

The main aim of the present paper is to carry out the systematic comparative study and assess the recently developed music-inspired harmony search (HS). The work presents the background of the problem, the solution strategies used, their implementation and the results obtained. The use of the algorithm, specific implementation, and a comparison of the performance are also presented.

2 Background: Harmony Search Algorithm

The detailed procedure of the proposed HS algorithm-based method [8] to determine optimal cross-sections in size optimization problems is shown in Fig. 1. The detailed procedure can be divided into the following two steps:

Initialization: HS algorithm parameters such as Harmony Memory Size (HMS), Harmony Memory Considering Rate (HMCR), Pitch adjusting Rate (PAR), maximum number of searches, number of x_i , and design variable bounds (member cross-sections A) are initialized. Harmonies (i.e., solution vectors) are then randomly generated from the possible variable bounds that are equal to the size of the HM. Here, the initial HM is generated based on a finite element method (FEM) structural analysis subjected to the constraint functions and sorted by the objective function values.

Search: A new harmony is improvised from the initially generated HM or possible variable values using the HMCR and PAR parameters. These parameters are introduced to allow the solution to escape from local optima and to improve the

global optimum prediction in the HS algorithm. The New Harmony is analyzed using the FEM method, and its fitness is evaluated using the constraint functions. If satisfied the weight of the structure is calculated using the objective function. If the New Harmony is better than the previous worst harmony, the New Harmony is included in the HM and the previous worst harmony is excluded from the HM. The HM is then sorted by the objective function value. The computations terminate when the maximum number of the search criterion is satisfied. If not, this step is repeated.

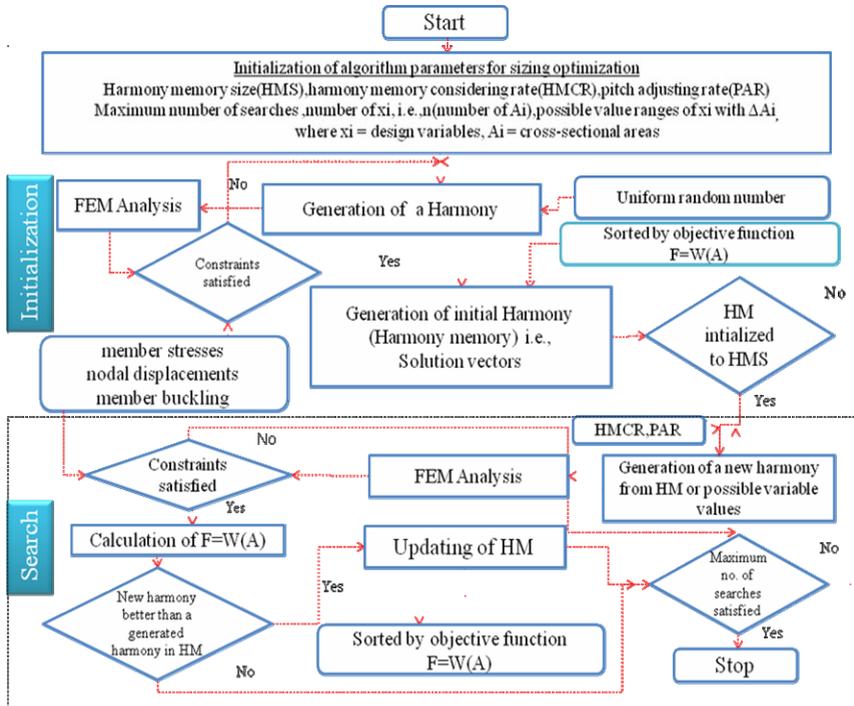


Fig. 1 Structural optimization design procedure using HS Algorithm [17]

2.1 Implementation of Algorithm

The first part of implementation is the development of a Truss Analysis component, which analyzes a truss and finds out and reports the members' stresses and node displacements. The truss analysis component is supposed to work for multiple load cases as well. The next three parts are implementation of the algorithms for the truss weight optimization problem [4,5].

2.2 Constraint Violation – Penalty Approach [24]

The fitness of a design vector also depends on whether it has violated any constraints defined in the problem. On the basis of that, it is ranked in comparison to other solutions in fitness. A common approach for including constraint violation into fitness is the method of penalty on the fitness value of the solution, for constraint violation, as per the strategy adopted. Some of the strategies are penalty proportional to absolute amount of total constraint violation, penalty proportional to number of total constraints violated, and so on. An inherent problem with all these approaches is that they do not adopt a strict approach towards discouraging solutions that violate constraints. While they adopt an approach to compensate the increase in area in feasible non-optimal solutions with the increase in penalty in non-feasible solution, the solution usually fails to converge in a strict feasible region, and the results are obtained with constraints violated, though the reduction in areas hides the effect due to a single summation. Here, a different approach to penalize the solutions, which violate the constraints, is adopted. An efficient constraint handling method proposed by Deb [24] is used in the current study. In this penalty function approach is used which does not require any penalty parameter.

Although a penalty term is added to the objective function to penalize infeasible solutions, the method differs from the way the penalty term is defined in conventional methods. The method proposes to use a tournament selection operator, where two solutions are compared at a time, and the following criteria are always enforced:

- Any feasible solution is preferred to any infeasible solution.
- Between two feasible solutions, the one having better objective function value is preferred.
- Between two infeasible solutions, the one having smaller constraint violation is preferred.

In the proposed method, penalty parameters are not needed because in any of the above three scenarios, solutions are never compared in terms of both objective function and constraint violation information. Of the three tournament cases mentioned above, in the first case, neither objective function value nor the constraint violation information is used; simply the feasible solution is preferred. In the second case, solutions are compared in terms of objective function values alone and in the third case; solutions are compared in terms of the constraint violation information alone.

Fitness function given below, where infeasible solutions are compared based on only their constraint violation.

$$F(\vec{x}) = \begin{cases} f(\vec{x}) & \text{if } g_j(\vec{x}) \geq 0 \quad \forall j = 1, 2, \dots, m, \\ f_{\max} + \sum_{j=1}^m \langle g_j(\vec{x}) \rangle & \text{otherwise.} \end{cases} \quad (1)$$

The parameter f_{max} is the objective function value of the worst feasible solution in the population. If no feasible solution exists in a population, f_{max} is set to zero. Thus, the fitness of an infeasible solution not only depends on the amount of constraint violation, but also on the population of solutions at hand. However, the fitness of a feasible solution is always fixed and is equal to its objective function value. It is observed that the above-mentioned rule set gives an optimum solution necessarily in the feasible region, and the solution obtained is optimal solution in the feasible region.

3 Numerical Example

Standard test case that has been used in previous truss size optimization papers was considered in this study. The MATLAB computer program was developed to demonstrate the efficiency and robustness of the HS algorithm. The structure is a 3D truss and contains 22 members and 8 nodes (Fig. 2).

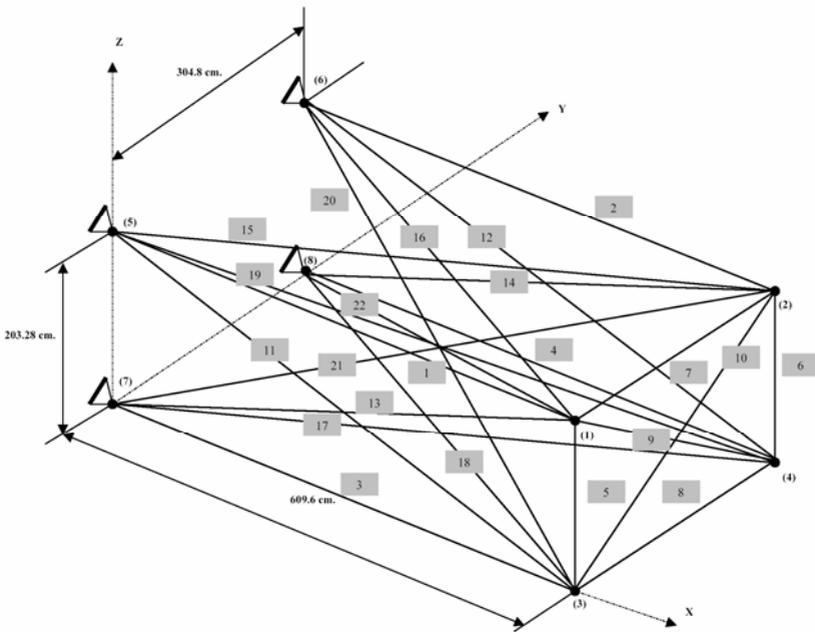


Fig. 2 Space truss

Four of the nodes were restrained to move totally and the truss is subjected to three loading conditions acting on the other four nodal points of the truss. These cases include truss structure subjected to three load conditions (Table 1).

Table 1 Loading Conditions for the 22-bar space truss

Node	Condition 1			Condition 2			Condition 3		
	P_x	P_y	P_z	P_x	P_y	P_z	P_x	P_y	P_z
1	-20	0	-5	-20	-5	0	-20	0	35
2	-20	0	-5	-20	-50	0	-20	0	0
3	-20	0	-30	-20	-5	0	-20	0	0
4	-20	0	-30	-20	-50	0	-20	0	-35

Note: loads are in kips, $1N = 0.2247\text{lbs}$.

The structure was analyzed using the FEM displacement method using following parameters.

- Elasticity modulus (E): 68947.6 MPa.
- Material Density (ρ): 27155.6 kN/m³.
- Constraints: displacement at any node: $-0.889\text{ cm} < \text{displacement} < 0.889\text{ cm}$.
- Grouping of members: The 22 members were linked into seven groups, two groups with five members each and another with four members (Table 2).
- Min. Cross sectional area: 0.06452 cm².
- The design variables are the cross-sectional areas of the individual truss elements and are continuous over the domain.
- The stress constraints are tabulated in Table 2 for differently grouped members.

Table 2 Member stress limitations for the 22-bar space truss

Variables	Compressive Stress Limitations (MPa)	Tensile Stress Limitations (MPa)
1 A1~A4	165.47	248.2
2 A5~A6	206.8	248.2
3 A7~A8	193.05	248.2
4 A9~A10	179.26	248.2
5 A11~A14	151.68	248.2
6 A15~A18	137.89	248.2
7 A19~A22	124.1	248.2

For the example presented in this study, the HS algorithm parameters were set as follows:

- Harmony memory size (HMS) = 20
- Harmony memory consideration rate (HMCR) = 0.8
- Pitch adjusting rate (PAR) = 0.3
- Maximum number of searches = 50,000

Objective function:

$$Z = \sum_{i=1}^m A_i L_i \rho_i \quad (2)$$

Where Z is total weight of the structure; m is total number of members, A is Cross-section area; L is length of the member and ρ is material density. Total no. of iterations taken were 2100. The results obtained are tabulated in Table 3. The HS algorithm has shown considerable success in optimizing the truss example used here.

Table 3 Comparative study

Variables	Optimal cross-sectional areas (cm. ²)			
	Sheu and Schmit [17]	Khan and Willmert [17]	Lee and Geem [17]	Present Study (HS and penalty approach)
A1 ~ A4	16.96	16.54	16.67	17.55
A5 ~ A6	7.497	10.02	6.99	7.039
A7 ~ A8	2.213	1.812	2.342	2.716
A9 ~ A10	2.729	3.303	2.72	2.717
A11 ~ A14	17.95	16.94	18.24	18.41
A15 ~ A18	14.02	13.75	13.26	12.90
A19 ~ A22	12.59	14.28	13.19	12.14
Wight (N)	4560.36	4604.593	4548.92	4544.8

Note: Weight in N, $1N = 0.2247\text{lbs}$, $1\text{cm.}^2 = 0.155\text{ in.}^2$

Some more examples of a 25-bar space truss subjected to two load conditions and a 72-bar space truss subjected to three load conditions are not included in the present paper, however the results can be found elsewhere [25]. HS algorithm performed differently in different cases, owing to a lot of factors. Based on these results, general discussion is given in following section.

4 General Discussion

- The Harmony search algorithm has proved to be the fastest and most efficient in finding near optimal solutions. Their convergence to global optimum is better than other heuristic algorithms.
- Harmony search algorithm is overall the best method for optimization of trusses as it generates a new vector after considering all the existing vectors based on Harmony Memory Considering Rate (HMCR) and the Pitch Adjusting Rate (PAR), rather than considering only (parents) as in genetic algorithms. These features increase the flexibility of the HS algorithm and produce better solutions.

- The algorithm tends to converge much more slowly for problems with higher degree of independent variables, due to increased complexity. Still the effect on actual solving also depends upon the complexity in variable interaction, which depends on the load and boundary conditions. For simpler loading cases, the convergence is much faster.
- The solutions obtained in the current work for Harmony search algorithm are all within the feasible space, and none of them violate the constraints. This is due the strict constraint violation approach followed here, which doesn't allow infeasible solutions to come in the top solutions, penalizing them during the ranking process.

5 Conclusion

The harmony search algorithm has been applied for weight optimization of spatial truss structure in the present paper. The results indicate that the harmony search algorithm is an efficient search and optimization method for solving the continuous sizing variables of the structures compared to conventional method in terms of both the obtained optimal solution and the convergence capability. Besides trusses, the HS algorithm can be applied to other types of structural optimization problems, including frame structures, plates, and shells.

References

1. Venkayya, V.B.: Design of optimum structures. *Comput. Struct.* 1, 265–309 (1971)
2. Patnaik, S.N.: Behavior of trusses with stress and displacement constraints. *Comput. Struct.* 22, 619–623 (1986)
3. Patnaik, S.N., Hopkins, D.A.: Optimality of a fully stressed design. *Comput. Methods Appl. Mech. Engg.* 165, 215–221 (1998)
4. Adeli, H., Kamal, O.: Efficient optimization of plane trusses. *Adv. Eng. Software* 13(3), 116–122 (1991)
5. Adeli, H., Cheng, N.T.: Integrated genetic algorithm for optimization of space structures. *J. Aerospace Eng., ASCE* 6(4), 315–328 (1993)
6. Ali, N., Behdinan, K., Fawaz, Z.: Applicability and viability of a GA based finite element analysis architecture for structural design optimization. *Comput. Struct.* 81, 2259–2271 (2003)
7. Deb, K., Gulati, S.: Design of truss-structures for minimum weight using genetic algorithms. *Finite Elements in Analysis and Design* 37, 447–465 (2001)
8. Geem, Z.W., Kim, J.H., Loganathan, G.V.: A new heuristic optimization algorithm: Harmony search. *Simulation. Comput. Struct.* 76, 60–68 (2001)
9. Geem, Z.W., Kim, J.H., Loganathan, G.V.: Harmony search optimization: Application to pipe network design. *International Journal of Modelling and Simulation* 22, 125–133 (2002)
10. Geem, Z.W., Tseng, C.L.: New Methodology, Harmony Search and Its Robustness. In: *Genetic and Evolutionary Computation Conference*, pp. 174–178 (2002)
11. Geem, Z.W.: Optimal Cost Design of Water Distribution Networks Using Harmony Search. *Engineering Optimization* 38(3), 259–280 (2006)

12. Geem, Z.W., Choi, J.Y.: Music Composition Using Harmony Search Algorithm. In: Giacobini, M. (ed.) *EvoWorkshops 2007*. LNCS, vol. 4448, pp. 593–600. Springer, Heidelberg (2007)
13. Geem, Z.W., Williams, J.C.: Harmony search and ecological optimization. *International Journal of Energy and Environment* 1, 150–154 (2007)
14. Geem, Z.W.: Optimal scheduling of multiple dam system using harmony search algorithm. In: Sandoval, F., Prieto, A.G., Cabestany, J., Graña, M. (eds.) *IWANN 2007*. LNCS, vol. 4507, pp. 316–323. Springer, Heidelberg (2007b)
15. Geem, Z.W.: Harmony search applications in industry. *Soft Computing Applications in Industry. Studies in Fuzziness and Soft Computing* 226, 117–134 (2008b)
16. Kim, J.H., Geem, Z.W., Kim, E.S.: Parameter Estimation of the Nonlinear Muskingum Model Using Harmony Search. *J. Am. Water Resour.Ass.* 37(5), 1131–1138 (2001)
17. Lee, K.S., Geem, Z.W.: A New Structural Optimization Method Based on the Harmony Search Algorithm. *Comput. Struct.* 82(9/10), 781–798 (2004)
18. Lee, K.S., Geem, Z.W.: A new meta-heuristic algorithm for continuous engineering optimization: Harmony search theory and practice. *Comput. Methods Appl. Mech. Eng.* 194, 3902–3933 (2005)
19. Lee, K.S., Geem, Z.W., Lee, S.-H., Bae, K.-W.: The harmony search heuristic algorithm for discrete structural optimization. *Eng. Optimiz* 37(7), 663–684 (2005)
20. Degertekin, S.O.: Optimum design of steel frames using harmony search algorithm. *Structural and Multidisciplinary Optimization* 7, 0177–4 (2007)
21. Erdal, F., Saka, M.P.: Effect of beam spacing in the harmony search based optimum design of grillages. *Asian Journal of Civil Engineering (Building and Housing)* 9, 215–228 (2008)
22. Ingram, G., Zhang, T.: *Overview of Applications and Developments in the Harmony Search Algorithm*, vol. 191, pp. 15–37. Springer, Heidelberg (2009)
23. Saka, M.P.: Optimum geometry design of geodesic domes using harmony search algorithm. *Advances in Structural Engineering* 10, 595–606 (2007)
24. Deb, K.: An efficient constraint handling method for genetic algorithms. *Comput. Methods Appl. Mech. Engg.* 186, 311–338 (2000)
25. Srikanth, D.: Comparative study of classical and evolutionary optimization techniques. M.Tech Dissertation, IIT Kharagpur (May 2009)

Nash Dominance with Applications to Equilibrium Problems with Equilibrium Constraints

Andrew Koh

Abstract. We examine a novel idea for the detection of Nash Equilibrium developed in [15] and apply it to Equilibrium Problems with Equilibrium Constraints (EPECs). EPECs are Nash games which uniquely feature players constrained by a condition governing equilibrium of a parametric system. By redefining the selection criteria used in evolutionary methods, EPECs can be solved using Evolutionary Multiobjective Optimization algorithms. We give a proposed algorithm (NDEMO) and illustrate it with numerical examples.

1 Introduction

A major trend in the provision of transportation services and facilities has been deregulation coupled with the private sector playing a larger role. When it occurs in highway [32] or transit [33], entities providing such services face competition from others with similar offerings. It is of interest to regulators to understand how such organizations make decisions on their service levels in this deregulated environment.

In this environment, the service levels provided are an outcome of a non-cooperative Nash game [18] amongst the players. However in transportation, this game possesses a feature that distinguishes it from the classic Nash game: The players' actions are constrained by a condition defining equilibrium in the transportation system [6]. In particular, the route choice decisions of users of a transportation network, which satisfy Wardrop's Equilibrium Principle [30], are parameterized in the decision variables of these firms. Therefore this is a hierarchical (i.e. leader-follower) game with the firms as leaders at the upper level engaged in a Nash game and travelers as followers at the lower level obeying an equilibrium condition. Thus the terms "firms", "leaders" and "players" are synonymous in this context.

Andrew Koh

Institute for Transport Studies, University of Leeds, Leeds, LS2 9JT, United Kingdom
e-mail: a.koh@its.leeds.ac.uk

The game described in the foregoing is an instance of a broader class of Equilibrium Problems with Equilibrium Constraints (or EPECs) ([16], [17]). EPECs have emerged as an area of research ([1], [29]) in mathematics applicable to transportation systems management and other disciplines ([9], [17]). This paper focuses on the determination of strategic variables for each profit maximizing leader when in competition with others. To avoid cumbersome notation, we implicitly assume henceforth that the equilibrium of the system acts as a binding constraint on the leaders.

Much of the game theory literature deals with games that are either zero sum where victory for one player is exactly balanced by the defeat of another (as in games such as checkers [2]) or where the actions of players are constrained to be in a discrete set (as in games such as Prisoner's Dilemma [24]). However the solution algorithms proposed for these are generally not applicable to EPECs. In this paper, we are concerned with games where the payoffs to the players are continuous and the strategic decision variables are subsets of the real line (as described in Chapter 6 of [31]). Our contribution is in the application of an evolutionary algorithm based on the proximity of solutions to a Nash Equilibrium (NE) for EPECs.

This paper is organized as follows. In the next section, the notions associated with the Nash game underlying the behavior of the leaders in the EPEC are introduced. Section 3 reviews deterministic (i.e. gradient based) and evolutionary approaches for computing NE in EPECs. Section 4 elucidates the Nash Domination criteria developed in [15] and provides an algorithm. Section 5 presents numerical examples of the solution of EPECs utilizing the concept of Nash Domination. Section 6 concludes the paper with a summary and directions for further research.

2 Nash Equilibrium

The leaders' problem in the EPEC is a single shot normal form game with a set of N players indexed by $i \in \{1, 2, \dots, n\}$ and each player can play a strategy $s_i \in S_i$ which all players are assumed to announce simultaneously. $S = \prod_{i=1}^n S_i$ is the collective action space for all players. It is convenient to denote s_{-i} as the combined strategies of all players in the game excluding that of player i i.e. $s_{-i} \equiv (s_1, \dots, s_{(i-1)}, s_{(i+1)}, \dots, s_n)$. So we have that $s \equiv (s_i, s_{-i})$ and we call s a strategy profile of all players in the game. Let $U_i(s)$ be the payoff to player $i, i \in N$ if s is played. Then a combined strategy profile $s^* = (s_1^*, s_2^*, \dots, s_n^*) \in S$ is a Nash Equilibrium for the game if the following holds:

$$U_i(s_i^*, s_{-i}^*) \geq U_i(s_i, s_{-i}^*) \quad \forall s_i \in S_i, \forall i \in \{1, 2, \dots, N\} \quad (1)$$

At a Nash Equilibrium no player can benefit (increase individual payoffs) by unilaterally deviating from her current strategy. As players do not cooperate, each player is doing the best she can taking into account what her competitors are doing [7]. Henceforth we refer to the NE problem as the determination of strategies that satisfy Equation 1.

3 Computation of Nash Equilibrium

3.1 Deterministic Approaches

In a game, the optimal move for a player is governed by her best response function. If $U_i(s)$ is continuously differentiable, then the best response function for player i is given by $\frac{dU_i(s_i, s_{-i})}{ds_i}$ ([7], [31]). The NE is the intersections of these best response functions for all players which amounts to finding solutions to N simultaneous equations i.e. solving $\frac{dU_i(s_i, s_{-i})}{ds_i} = 0, \forall i \in \{1, 2, \dots, N\}$.

While useful for providing insights into the behavior of players, the analytical method is not feasible for realistic problems and even less so for EPECs due to the binding equilibrium condition. Thus the practical approach for finding NE is by using variants of fixed point iteration (e.g. nonlinear Gauss-Siedel) ([10], [29]) or by formulating it as a Complementarity Problem ([1]). Applications of these methods are found in (e.g. [8], [14]). Convergence of these algorithms rely on the payoff functions being continuously differentiable and possessing diagonally dominant Jacobians ([7], Theorem 4.1, pp. 280). However, if the payoff functions of the players are not concave, there may exist NE that satisfy Equation 1 locally but not globally. This is known as a “local NE trap” ([28], Definition 3, pp.306). There is thus a parallel with the literature on multimodal function optimization where the potential for multiple optima cannot be ignored. Thus apart from their differentiability requirements, another drawback of deterministic approaches is that they can fall prey to the local NE trap, an occurrence crucially dependent on the starting point used in these algorithms. For details of these and other deterministic methods, see [6].

3.2 Evolutionary Methods

Due to the ability of evolutionary algorithms in dealing with non-smooth and non-differentiable functions and their reported success in escaping local optima and potentially a local NE trap, evolutionary counterparts of deterministic fixed point iteration methods were proposed in ([23], [25], [27]).

Another strand of research has been the exploitation of coevolution since it was first demonstrated in tackling multi-dimensional function optimization (e.g. [20]). Several subpopulations (one representing each problem dimension) are evolved simultaneously to avoid premature convergence and to widen the search of the problem space. Ideas from coevolution have been exported into algorithms designed for the detection of NE; here each subpopulation encodes the strategies of individual players ([4], [19], [22]). However doubts have been cast on the performance of coevolutionary methods. In [28], the coevolutionary algorithm had to be hybridized with local search techniques to enable successful detection of NE. [12] developed a coevolutionary particle swarm optimization method which attempted to detect the NE by learning the best response functions of the players. Instead of using the coevolutionary paradigm of previous works, a novel idea exploiting the concept of Nash Dominance was proposed [15] to find NE as discussed in Section 4.

4 Nash Domination

At their most abstract level, evolutionary multiobjective (EMO) algorithms ([3], [5]) apply stochastic operators to a parent population with the aim of evolving a fitter child population to solve vector valued optimization problems. Subsequently, in the selection phase, a comparison is made between a chromosome x from the parent population and a chromosome y from the child population on the basis of fitness and the weaker of the two is discarded. Given that one of the tasks in EMO is to identify the entire Pareto frontier [5], fitness is assigned based on Pareto Domination (PD): x Pareto Dominates y if x is strictly no worse off than y in all objectives *and* x is better than y in at least one objective ([5], Definition 2.5, pp. 28).

[15] define a concept analogous to PD called Nash Domination for the NE problem. A chromosome here represents the strategies of all N players concatenated into a vector i.e. a strategy profile. Then instead of using PD to compare two chromosomes i.e. two strategy profiles, Nash Domination operates by counting the number of players that can benefit if each player switches strategies *in turn*. The *fewer* the number of players that can benefit by deviating from one profile compared to the other, the closer the former is to a NE as defined in Section 2.

Consider two strategy profiles $\{a, b\} \in S$, ($a \equiv (a_1, \dots, a_n)$, $b \equiv (b_1, \dots, b_n)$), and define an operator $k : S \times S \rightarrow N$ associating the cardinality of a set defined by 2:

$$\{i \in \{1, \dots, n\} \mid U_i(b_i, a_{-i}) \geq U_i(a), b_i \neq a_i\} \quad (2)$$

This set defined by (2) comprises the players that would benefit by playing b_i when everyone else plays a_{-i} . The total number of players in this set is given by $k(a, b)$. A similar interpretation applies, *mutatis mutandis*, for $k(b, a)$. Note that to evaluate $k(a, b)$ and $k(b, a)$, the payoff to each player, individually, from deviating has to be computed. Then in a pairwise comparison of two strategy profiles, either one of the following must be true: ([15], Remark 4, pp. 365)

1. $k(a, b) < k(b, a) \rightarrow a$ Nash Dominates b or
2. $k(b, a) < k(a, b) \rightarrow b$ Nash Dominates a or
3. $k(a, b) = k(b, a) \rightarrow a$ and b are Nash Non Dominated (NND) with respect to each other.

It has been proven ([15], Proposition 9, pp. 366) that all NND chromosomes are NE. Thus instead of checking for PD when comparing chromosomes, an EMO algorithm can be modified to identify strategy profiles that are NND. Then rather than locating the Pareto Frontier, we collect its analogue: the Non Nash Dominated Frontier to which the population converges. A proposed Nash Domination Evolutionary Multiplayer Optimization (NDEMO) algorithm is given in Algorithm 1. NDEMO is based on the method of [26] which relies on Differential Evolution (DE) [21]. By modification of this selection criteria, any other EMO algorithm (see [3] or [5] for alternatives) can be used.

NDEMO operates as follows. The user specifies the maximum number of generations Max_{it} , the population size h , the convergence criteria, $\epsilon (> 0)$, control parameters required in DE [21] and a procedure to compute payoffs. Initial parent strategy

Algorithm 1. Nash Domination Evolutionary Multiplayer Optimization (NDEMO)

Input: h, Max_{it}, ε , DE Control Parameters, payoff functions
 $it \leftarrow 0$
Randomly initialize parent strategy profiles \mathcal{P}
Evaluate payoffs to players with \mathcal{P}
while $it < Max_{it}$ or \mathcal{P} not converged **do**
 Apply DE operators to create child strategy profiles \mathcal{C} :
 $\mathcal{C} \stackrel{DE}{\leftarrow} \mathcal{P}$
 Evaluate payoffs to players with \mathcal{C}
 Perform Pairwise Nash Domination Comparison between \mathcal{P} and \mathcal{C} :
 $\mathcal{T} \leftarrow \emptyset$
 for $j = 1$ to h **do**
 $a \leftarrow \mathcal{P}_j^{it}$
 $b \leftarrow \mathcal{C}_j^{it}$
 if $k(a, b) < k(b, a)$ **then**
 reject b
 $\mathcal{T} \leftarrow a$
 else if $k(b, a) < k(a, b)$ **then**
 reject a
 $\mathcal{T} \leftarrow b$
 else
 $\mathcal{T} \leftarrow a$
 $\mathcal{T} \leftarrow b$
 end if
 end for
 if size of $\mathcal{T} > h$ **then**
 Randomly trim \mathcal{T} until h remain
 end if
 $\mathcal{P}^{(it+1)} \leftarrow \mathcal{T}$
 Convergence Check:
 Randomly choose a chromosome m from \mathcal{P}^{it+1}
 Compute Euclidean distance norm between m and every other member in \mathcal{P}^{it+1}
 if maximum of norm $\leq \varepsilon$ **then**
 Terminate
 else
 $it \leftarrow it + 1$
 end if
end while
Output: Nash Non Dominated Solutions

profiles \mathcal{P} are generated randomly. Then child strategy profiles \mathcal{C} are created by applying the DE operators via the stochastic combination of randomly chosen parents as discussed in [21]. At each generation, parent and child strategy profiles are compared one by one pairwise, following the Nash Domination procedure described previously. Those chromosomes that are NND are stored in a temporary population \mathcal{T} . However, this means that the size of \mathcal{T} can exceed h . In that case, we randomly trim \mathcal{T} so that there will always be only h parents for the next generation.

As a check on convergence to the NE, we randomly choose a chromosome m and compute the Euclidean distance between m and every other member of \mathcal{P} . If the maximum distance is less than ε , the population is deemed to have converged to an NE. The algorithm is then repeated until either Max_{it} is reached or until convergence is achieved.

5 Numerical Examples

The examples presented are typical of situations when a private profit maximizing firm competes with others in the operation of private roads. The interaction between these firms and users of the highway network is depicted in Figure 1. To explicitly account for the hierarchical nature of the game and that route choices of users on the network must satisfy Wardrop's Equilibrium Condition [30], a traffic assignment problem is solved for a given strategy profile of the players to obtain the traffic flows which each player's payoffs depend on.

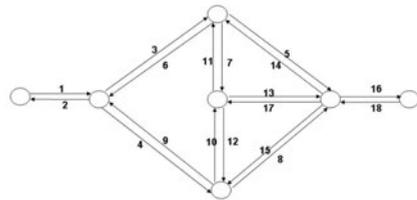
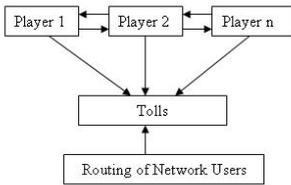


Fig. 1 Hierarchical Game with Equilibrium Route Choice

Fig. 2 Highway Network with 18 one way roads from [13] (links labeled are road numbers)

This example is taken from [13]. There are two players and each firm chooses toll levels (one firm per road) on the network (see Figure 2) to maximize toll revenues (given as the product of tolls and traffic flows). Assume firstly that roads 7 and 10 are the only tolled roads. This example was solved as a complementarity problem in [13] and via a Coevolutionary Particle Swarm Algorithm in [12]. We used a population size of 20 chromosomes, DE control parameters from [26] and terminate the algorithm when $\varepsilon \leq 1e - 4$. NDEMO took 38 minutes to converge to the tolls (in seconds) shown in Table 1 which agrees with previous results.

Next we consider the situation when, in addition to roads 7 and 10 being tolled, another player maximizes profits by charging tolls on road 17. The results are reported in Table 2. Although NDEMO again successfully converged to the NE (as verified by solving it as a complementarity problem following the method described in [13]), this time NDEMO took 54 minutes to meet the same convergence criteria. Thus with one additional player, the time taken has increased by 42% over the 2 player case. The increase in computing time stems from the domination checking procedure combined with the hierarchical nature of the game. This translates into

Table 1 Tolls (seconds) in Two Player Model

Firm	Road	NDEMO [12]	[13]
1	7	141.36	141.36 141.37
2	10	138.28	138.29 138.29

Table 2 Tolls (seconds) in Three Player Model

Firm	Road	NDEMO	Method of [13]
1	7	140.94	140.96
2	10	137.51	137.56
3	17	711.25	712.88

the requirement to calculate the profits to each firm from deviating (so as to obtain $k(a, b)$ and $k(b, a)$) by solving a traffic assignment problem for each player in turn.

6 Conclusions

In this paper, we proposed modifying an EMO algorithm for solving EPECs by extending the procedure suggested in [15]. This revised algorithm (NDEMO) enabled us to handle Nash games where players encounter a system equilibrium constraint. Numerical examples illustrating competition in private sector provision of highway transportation were given to demonstrate the performance of the proposed algorithm. While the examples suggest that this could be a potentially useful method for EPECs, we stress the need, in the pairwise comparison, to compute the payoff to each player, one by one, from deviating. This implies that the computational complexity of NDEMO increases significantly as the number of players increase as evidenced by the increase in computational times required in our examples.

An area of further research would be the effects of the control parameters of NDEMO on the speed of convergence to NND solutions since we have used parameters suggested in [26]. NDEMO could also be extended to other domains where EPECs are applicable such as electricity markets (e.g. [9]). Additionally, comparisons with other algorithms are currently being undertaken.

References

1. Červinka, M.: Hierarchical structures in equilibrium problems. PhD Thesis, Charles University, Prague, Czech Republic (2008)
2. Chellapilla, K., Fogel, D.: Evolving neural networks to play checkers without expert knowledge. *IEEE Transactions on Neural Networks* 10(6), 1382–1391 (1999)
3. Coello-Coello, C., Lamont, G.: Applications of multi-objective evolutionary algorithms. World Scientific, Singapore (2004)
4. Curzon Price, T.: Using co-evolutionary programming to simulate strategic behavior in markets. *Journal of Evolutionary Economics* 7(3), 219–254 (1997)
5. Deb, K.: Multi-objective optimization using evolutionary algorithms. John Wiley, Chichester (2001)
6. Facchinei, F., Kanzow, C.: Generalized Nash equilibrium problems. *4OR* 5(3), 173–210 (2007)

7. Gabay, D., Moulin, H.: On the uniqueness and stability of Nash-equilibria in non cooperative games. In: Bensoussan, A., et al. (eds.) *Applied Stochastic Control in Econometrics and Management Science*, pp. 271–293. North Holland, Amsterdam (1980)
8. Harker, P.T.: A variational inequality approach for the determination of Oligopolistic market equilibrium. *Mathematical Programming* 30(1), 105–111 (1984)
9. Hu, X., Ralph, D.: Using EPECs to model bilevel games in restructured electricity markets with locational prices. *Operations Research* 55(5), 809–827 (2007)
10. Judd, K.: *Numerical methods in Economics*. MIT Press, Cambridge (1998)
11. Karamardian, S.: Generalized complementarity problems. *Journal of Optimization Theory and Applications* 8(3), 161–168 (1971)
12. Koh, A.: Coevolutionary particle swarm algorithm for bi-level variational inequalities: applications to competition in highway transportation networks. In: Chiong, R., Dhakal, S. (eds.) *Natural intelligence for scheduling, planning and packing problems*, pp. 195–217. Springer, Berlin (2009)
13. Koh, A., Shepherd, S.: Tolling, collusion and equilibrium problems with equilibrium constraints. *European Transport/Trasporti Europei* (in press)
14. Kolstad, M., Mathisen, L.: Computing Cournot-Nash equilibrium. *Operations Research* 39(5), 739–748 (1991)
15. Lung, R.I., Dumitrescu, D.: Computing Nash equilibria by means of evolutionary computation. *International Journal of Computers, Communications and Control* III, 364–368 (2008)
16. Mordukhovich, B.S.: Optimization and equilibrium problems with equilibrium constraints. *Omega* 33(5), 379–384 (2005)
17. Mordukhovich, B.S.: *Variational analysis and generalized Differentiation, II: Applications*. Grundlehren der mathematischen wissenschaften, vol. 331. Springer, Berlin (2006)
18. Nash, J.: Equilibrium points in N-person games. *Proceedings of the National Academy of Science USA* 36(1), 48–49 (1950)
19. Pedroso, J.P.: Numerical solution of Nash and Stackelberg equilibria: an evolutionary approach. In: *Proceedings of SEAL 1996*, pp. 151–160 (1996)
20. Potter, M.A., De Jong, K.: A cooperative coevolutionary approach for function optimization. In: *Proceedings of PPSN III*, pp. 249–257. Springer, Berlin (1994)
21. Price, K., Storn, R., Lampinen, J.: *Differential evolution: a practical approach to global optimization*. Springer, Berlin (2005)
22. Protopapas, M., Kosmatopoulos, E.: Determination of sequential best replies in n-player games by genetic algorithms. *International Journal of Applied Mathematics and Computer Science* 5(1), 19–24 (2009)
23. Rajabioun, R., Atashpaz-Gargari, E., Lucas, C.: Colonial competitive algorithm as a tool for Nash equilibrium point achievement. In: Gervasi, O., Murgante, B., Laganà, A., Taniar, D., Mun, Y., Gavrilova, M.L. (eds.) *ICCSA 2008, Part II. LNCS*, vol. 5073, pp. 680–695. Springer, Heidelberg (2008)
24. Rapoport, A., Chammah, A.: *Prisoner's Dilemma*. University of Michigan Press, Ann Arbor (1965)
25. Razi, K., Shahri, S.H., Kian, A.R.: Finding Nash equilibrium point of nonlinear non-cooperative games using coevolutionary strategies. In: *Proceedings of ISDA*, pp. 875–882 (2007)
26. Robič, T., Filipič, B.: DEMO: differential evolution for multiobjective problems. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) *EMO 2005. LNCS*, vol. 3410, pp. 520–533. Springer, Heidelberg (2005)
27. Sefrioui, M., Periaux, J.: Nash genetic algorithms: examples and applications. In: *Proceedings of IEEE CEC*, pp. 509–516 (2000)

28. Son, Y., Baldick, R.: Hybrid coevolutionary programming for Nash equilibrium search in games with local optima. *IEEE Transactions on Evolutionary Computation* 8(4), 305–315 (2004)
29. Su, C.: Equilibrium problems with equilibrium constraints: stationarities, algorithms and applications. PhD Thesis, Stanford University, California, USA (2005)
30. Wardrop, J.G.: Some theoretical aspects of road traffic research. *Proceedings of Institution of Civil Engineers Part II* 1(36), 325–378 (1952)
31. Webb, J.N.: *Game theory: decisions, interaction and Evolution*. Springer, London (2007)
32. Yang, H., Feng, X., Huang, H.: Private road competition and equilibrium with traffic equilibrium constraints. *Journal of Advanced Transportation* 43(1), 21–45 (2009)
33. Zubeita, L.: A network equilibrium model for oligopolistic competition in city bus services. *Transportation Research Part B* 32(6), 413–422 (1998)

A New Evolutionary Based Approach for Solving the Uncapacitated Multiple Allocation p-Hub Median Problem

Marija Milanović

Abstract. In this paper a new evolutionary based approach for solving the uncapacitated multiple allocation p-hub median problem is presented. Integer encoding of individuals is used, which provided the feasibility of individuals and enabled the usage of standard genetic operators. The experiments were carried out on the standard ORLIB AP data set. The proposed algorithm achieved three new best known solutions.

1 Introduction

The problem of locating hub facilities arises frequently in the design of communication networks, as well as in airline passenger flow and parcel delivery networks.

There are several types of hub network problems ([4, 5]). In this paper, the uncapacitated multiple allocation p-hub median problem (UMApHMP) is considered. The problem is “uncapacitated” which means that there are no capacity limitations on the hubs or on the flow between arcs. “Multiple allocation” allows each non-hub node to be allocated to several or all of the hubs in such a way that the overall cost of satisfying the flow demand is minimized. The assumption is that the hubs are fully interconnected while non-hub nodes must route all their traffic indirectly via one or more hubs. The number of hubs must be exactly equal to p . Setup costs are ignored.

It is assumed that there are known non-negative flows associated with each origin-destination pair of nodes. The operating cost of a network depends on the distance between nodes in the network, the amount of flow to be moved across these distances and the type of the link between the nodes (whether the flow is translated between hubs, collected from or distributed to a non-hub node). The cost coefficients corresponding to these types of links are, respectively, denoted as χ, α, δ .

Marija Milanović

University of Belgrade, Faculty of Mathematics, Studentski trg 16/IV,

11000 Belgrade, Serbia

e-mail: marija.milanovic@gmail.com

The problem is known to be NP-hard, with exception of the special cases (for example, sparse matrix of flows) that are solvable in polynomial time. If the set of hubs is fixed, the problem can also be polynomially solved using the shortest-path algorithm in $O(n^2p)$ time.

2 Previous Work

In 1987, O’Kelly first formulated the discrete hub location problem, while Skorim-Kapon et. al. (1996) first computed the optimal solutions of CAB data set by CPLEX solver. In 1998, Ernst and Krishnamoorthy [6] used the shortest path method to enumerate all possible hub locations for the UMAPHMP.

Some of the recently published papers devoted to the UMAPHMP are: [3, 11, 8]. In [3], Boland et. al. developed a heuristic and an exact Branch-and-Bound method for solving multiple allocation hub location problems by using pre-processing and cutting algorithms. They first obtain good upper bounds that are used to cut the size of branch and bound tree, but this approach gives results only on CAB ($n \leq 25$, $p \leq 4$) and smaller size AP instances ($n \leq 50$, $p \leq 5$). Stanimirović [11] solved the UMAPHMP by using genetic algorithm with shortest-path objective function, binary representation of the individuals, and modified genetic operators adjusted to that representation. The algorithm was tested on the standard ORLIB hub instances (CAB and AP) and gave results on the API instances that were not considered in the literature before. In [8] authors developed modified ant colony optimization (ACO) algorithm for the UMAPHMP. Hub location selection and non-hub nodes allocation were considered in solution constructions, and a local search mechanism was adopted to enhance the quality of solutions in shorter computation time. Two multiple allocation policies were proposed, nearest hub and all pairs shortest path. The results showed that the (ACO) can solve larger instances with small CPU time increase in comparison with published results.

3 Mathematical Representation

In this paper, a mixed integer linear programming formulation of the UMAPHMP from [11], which was introduced in [3], is used.

Consider a set $I = \{1, 2, \dots, n\}$ of n distinct nodes in the network, where each node refers to origin/destination or potential hub location. The distance from node i to node j is C_{ij} , and the triangle inequality may be assumed. The demand from origin i to destination j is denoted with W_{ij} . The number of hubs to be located is fixed to p . Decision variables H_j , Z_{ik} , Y_{kl}^i , and X_{lj}^i are used in the formulation as follows:

- $H_j = 1$, if a hub is located at node j , 0 otherwise
- $Z_{ik} =$ the amount of flow from node i that is collected at hub k
- $Y_{kl}^i =$ the amount of flow from node i that is collected at hub k , and transported via hub l
- $X_{lj}^i =$ the amount of flow from node i to destination j that is distributed via hub l

Parameters χ and δ denote unit costs of collection and distribution, while $1 - \alpha$ represents a discount factor for the transportation costs between hubs. The objective is to locate exactly p hub facilities, such that the total flow cost is minimized. Using the notation mentioned above, the problem can be written as:

$$\min \sum_i [\chi \sum_k C_{ik} Z_{ik} + \alpha \sum_k \sum_l C_{kl} Y_{kl}^i + \delta \sum_l \sum_j C_{lj} X_{lj}^i] \quad (1)$$

subject to:

$$\sum_i H_i = p \quad (2)$$

$$\sum_k Z_{ik} = \sum_j W_{ij} \text{ for every } i \quad (3)$$

$$\sum_l X_{lj}^i = W_{ij} \text{ for every } i, j \quad (4)$$

$$\sum_l Y_{kl}^i + \sum_j X_{kj}^i - \sum_l Y_{lk}^i - Z_{ik} = 0 \text{ for every } i, k \quad (5)$$

$$Z_{ik} \leq \sum_j W_{ij} H_k \text{ for every } i, k \quad (6)$$

$$\sum_i X_{lj}^i \leq \sum_i W_{ij} H_l \text{ for every } l, j \quad (7)$$

$$X_{lj}^i, Y_{kl}^i, Z_{ik} \geq 0, H_k \in \{0, 1\} \text{ for every } i, j, k, l \quad (8)$$

The objective function (1) minimizes the sum of origin-hub, hub-hub and hub-destination flow costs multiplied with χ , α and δ factors respectively. Constraint (2) limits the number of located hubs to p , while (3)-(5) represent the divergence equations for the network flow problem for each node i . Constraints (6) and (7) unable direct communication between non-hub nodes, while (8) reflects non-negative and/or binary representation of decision variables.

4 Proposed Evolutionary Algorithm

Evolutionary algorithms (EA) are a class of optimization algorithms. EAs solve problems by creating a population consisting of individuals. An individual represents an encoded solution of the problem. Individuals from the current population are evaluated by using a fitness function to determine their qualities. By applying genetic operators of selection, crossover, and mutation on the current generation, the next generation of individuals is produced. The process is repeated until stopping criterion is satisfied.

In [11], Stanimirović used a binary encoding of individuals. Because of the problem with the preservation of feasibility of individuals, genetic operators needed to be modified.

Here, an integer encoding of individuals is used. The genetic code of each individual consists of a string of length p where the first gene of the string, $gene_1$, can take a value from the set $\{1, 2, \dots, n\}$, the second gene, $gene_2 \in \{1, 2, \dots, n-1\}$, ..., and, finally, the p th gene, $gene_p \in \{1, 2, \dots, n-p+1\}$. These values determine the indices of established hub locations in the following way: the i -th established hub is the $gene_i$ -th unused potential hub location. Thus, index of the first established hub is exactly equal to the value of the first gene, because no hubs were opened earlier. Then, index of the second established hub is obtained as $gene_2$ -th node which is not already established as a hub, and so on.

Example Let $n=6$, $p=4$, and genetic code be 2112. Index of the first established hub is 2 (the first 2 in the genetic code). Index of the second established hub is 1 because it is the first unused potential location (the first 1 in the genetic code). Then, used locations are those with indices 1 and 2, and index of the third established hub is 3 because it is now the first which is not already used (the second 1 in the genetic code). The fourth hub to be established is the second unused (2 in the genetic code), which is actually potential location with index 5 (because locations with indices 1, 2, and 3 are used up to now).

In the used implementation with integer encoding, all individuals are always feasible, thus standard genetic operators are used. Integer encoding introduced an epistasis. More on the epistasis can be found in [10].

Like in [11], for a fixed set of hubs (H_j), the objective value of an individual is calculated using a well known Floyd-Warshall shortest path algorithm, described in [1, 6].

Fine grained tournament selection (FGTS) proposed in [7] with parameter $F_{tour} = 5.4$ is used. Also, standard one-point crossover operator is applied, with the rate of $p_{cross} = 0.85$. Thus, around 85% pairs of individuals previously chosen by operator of selection participate in producing offspring. Applied crossover operator exchanges segments of two selected individual's genetic code after the randomly chosen position. Operator of simple mutation is used, which causes random changes by modifying individual's genetic material with basic mutation rate of $\frac{0.4}{n}$ for non-frozen bits, and 2.5 times higher mutation rate on frozen ones. If, during the execution of algorithm, it happens that on the certain position all individuals in the population have the same value of the genes, that position is called frozen gene. Frozen genes significantly decrease the search space and can cause a premature convergence. Since the selection and crossover operators cannot change a value of the frozen gene, while mutation can, but its probability is quite small, the probability of mutation is increased on frozen genes.

In [12] GA parameters were intensively tested and values reported as best are chosen here.

The population has $N_{pop} = 150$ individuals, including $N_{elite} = 100$ of elitist ones. Elitist individuals pass directly to the next generation, without applying genetic operators on them. Individuals in the first generation are randomly generated. Duplicated individuals are removed from the population in the next generation by setting their fitness value to zero. Number of individuals with the same objective value, but

different genetic codes, is limited to $N_{rv} = 40$. If the number of such individuals become too large, it can yield to the local optimum.

The running time of GA is improved by caching technique in a way described in [9].

5 Computational Results

All experiments were executed on Intel 2.5 GHz PC with 1GB RAM under Windows XP operating system. The algorithm was coded in C programming language.

The AP data set of standard ORLIB [2] test problems was used to evaluate the computational characteristics of the evolutionary algorithm.

The stopping criterion of the algorithm was the maximum number of generations equal to 5,000 or at most 2,000 successive generations without any improvement of the objective value. GA was run 20 times on each instance.

Table 1 contains results on the AP instances for which an optimal solution is known. The table is organized as follows:

- the first column contains the number n of nodes;
- in the second column the number p of hubs to be established is presented;
- optimal solutions are given in the third column, while in the fourth the best EA solutions are presented. If the best EA solution is equal to the optimal solution, the mark *opt* is written in the fourth column;
- the average running time (t) used to reach the final EA solution for the first time is given in the fifth column, while the sixth and seventh columns (t_{tot} and gen) show the average total running time and the average number of generations for finishing EA, respectively. Note that running time t_{tot} includes t ;

In both Table 1 and Table 2 last two columns contain t_{tot} and gen values of the GA implementation from [11].

Table 2 contains current best known solutions on the AP instances, based on results from [11] and new results of EA. The table is organized as follows:

- first three columns represent the same type of values as in Table 1;
- in the fourth column previously best known solution is given;
- the fifth column contains the best EA solution. If the best EA solution is equal to the previously best known solution, the mark *best* is written;
- next three columns (named with t , t_{tot} , and gen) represent the same type of values as in Table 1
- New best solutions achieved by EA are marked with *new best!*.

From Table 1 it can be seen that EA reached all optimal solutions. On the instances where optimal solution is not known (Table 2), EA reached all best known solutions from the literature, except on three instances ($n=100, p=20$; $n=200, p=15$; $n=200, p=20$). On other three instances ($n=200, p=6$; $n=200, p=7$; $n=200, p=9$) EA achieved new best known solutions.

Table 1 Optimal Solutions on the AP instances

n	p	OPT_{sol}	GA_{best}	t (sec)	t_{tot} (sec)	gen	prevGA	
							t_{tot}	gen
10	2	163603.94	<i>opt</i>	0.000	0.013	201.0	0.037	201
10	3	131581.79	<i>opt</i>	0.001	0.025	201.0	0.038	201
10	4	107354.73	<i>opt</i>	0.001	0.046	201.0	0.040	204
10	5	86028.88	<i>opt</i>	0.001	0.058	201.0	0.042	201
10	6	72427.73	<i>opt</i>	0.001	0.072	201.0	0.042	201
10	7	63466.81	<i>opt</i>	0.001	0.080	201.0	0.041	202
10	8	54628.75	<i>opt</i>	0.003	0.086	201.0	0.041	202
20	2	168599.79	<i>opt</i>	0.001	0.025	202.7	0.045	201
20	3	148048.30	<i>opt</i>	0.004	0.075	208.7	0.065	210
20	4	131665.43	<i>opt</i>	0.005	0.120	212.2	0.091	213
20	5	118934.97	<i>opt</i>	0.011	0.160	216.2	0.119	210
20	6	107005.85	<i>opt</i>	0.012	0.207	211.7	0.161	226
20	7	97697.75	<i>opt</i>	0.022	0.249	217.4	0.184	209
20	8	91454.83	<i>opt</i>	0.030	0.299	219.5	0.211	227
25	2	171298.10	<i>opt</i>	0.002	0.031	201.0	0.051	201
25	3	151080.66	<i>opt</i>	0.003	0.090	206.3	0.088	209
25	4	135638.58	<i>opt</i>	0.006	0.145	209.6	0.139	212
25	5	120581.99	<i>opt</i>	0.040	0.227	238.7	0.208	223
25	6	110835.82	<i>opt</i>	0.064	0.320	243.8	0.277	246
25	7	103880.23	<i>opt</i>	0.106	0.417	259.6	0.374	257
25	8	97795.59	<i>opt</i>	0.124	0.473	253.2	0.453	252
40	2	173415.96	<i>opt</i>	0.007	0.087	206.9	0.102	211
40	3	155458.61	<i>opt</i>	0.025	0.259	217.8	0.245	226
40	4	140682.74	<i>opt</i>	0.066	0.392	234.9	0.452	234
40	5	130384.74	<i>opt</i>	0.127	0.547	252.8	0.788	299
40	6	122171.26	<i>opt</i>	0.190	5.856	2060.9	4.834	2039
50	2	174390.03	<i>opt</i>	0.017	0.134	210.3	0.173	228
50	3	156014.72	<i>opt</i>	0.003	0.341	201.0	0.511	288
50	4	141153.38	<i>opt</i>	0.098	0.551	234.6	0.885	285
50	5	129412.60	<i>opt</i>	0.186	0.787	254.2	1.282	301
50	6	121671.76	<i>opt</i>	0.356	8.479	2079.1	9.150	2284
50	10	100508.95	<i>opt</i>	2.841	17.985	2366.1	21.136	2412
100	2	176245.38	<i>opt</i>	0.131	5.144	2020.6	2.736	2089
100	3	157869.93	<i>opt</i>	0.489	13.317	2062.9	13.227	2207
100	4	143004.31	<i>opt</i>	1.425	18.516	2155.6	32.848	2652
100	5	133482.57	<i>opt</i>	1.764	23.778	2147.9	54.389	3097
200	2	178093.99	<i>opt</i>	0.952	44.161	2030.1	35.686	2129
200	3	159725.11	<i>opt</i>	3.994	73.947	2103.5	174.900	2520

Table 2 Best Known Solutions on the AP instances

n	p	<i>old</i>	EA_{best}	t	t_{tot}	gen	prevGA	
		$BEST_{sol}$					(sec)	(sec)
40	7	116036.38	<i>best</i>	0.445	7.122	2125.8	6.002	2086
40	8	109971.92	<i>best</i>	0.553	8.388	2135.6	7.655	2085
40	9	104212.42	<i>best</i>	0.507	9.625	2104.2	9.010	2127
40	10	99452.67	<i>best</i>	2.063	12.483	2388.1	9.491	2085
50	7	115911.64	<i>best</i>	0.597	10.338	2114.1	15.725	2851
50	8	109926.60	<i>best</i>	1.032	12.515	2171.8	17.188	2591
50	9	104968.27	<i>best</i>	1.773	15.166	2261.5	17.252	2298
50	11	96186.22	<i>best</i>	4.949	22.929	2544.4	24.675	2454
50	12	93171.96	<i>best</i>	6.502	25.959	2673.4	23.870	2267
50	13	90409.79	<i>best</i>	6.610	27.884	2618.7	27.221	2281
50	14	87654.61	<i>best</i>	9.542	32.361	2798.5	29.098	2238
50	15	85032.89	<i>best</i>	9.442	34.576	2754.6	35.493	2456
50	20	73490.33	<i>best</i>	20.201	52.888	3149.8	39.859	2094
100	6	126107.56	<i>best</i>	3.371	31.270	2231.1	99.973	4350
100	7	120165.15	<i>best</i>	9.201	41.323	2555.8	100.118	3553
100	8	114295.92	<i>best</i>	18.449	57.004	2881.0	125.793	3891
100	9	109448.87	<i>best</i>	22.015	68.737	2926.4	126.037	3409
100	10	104794.05	<i>best</i>	36.274	87.160	3303.5	146.263	3421
100	15	88882.05	<i>best</i>	81.364	167.069	3735.7	270.956	4004
100	20	79191.02	79245.53	118.613	233.860	3853.8	377.160	3828
200	4	144508.20	<i>best</i>	10.586	95.512	2239.4	376.815	3585
200	5	136761.83	<i>best</i>	55.526	156.214	3038.6	562.245	4231
200	6	129560.60	129556.47	68.761	185.721	3108.4	681.338	4281 <i>new best!</i>
200	7	123609.44	123608.87	89.739	226.298	3306.8	766.016	4219 <i>new best!</i>
200	8	117709.98	<i>best</i>	128.293	284.977	3559.8	879.377	4237
200	9	112380.66	112374.47	206.524	366.924	3940.0	1096.180	4809 <i>new best!</i>
200	10	107846.82	<i>best</i>	258.383	432.686	4106.1	1157.049	4591
200	15	92669.64	92810.67	574.189	816.383	4720.2	1750.105	4699
200	20	83385.94	83951.07	914.171	1130.005	4882.9	2425.588	4924

6 Conclusions

A new evolutionary metaheuristic for solving the UMAPHMP is described. Integer encoding of individuals is used which provided the feasibility of individuals all the time and also the usage of standard genetic operators. The idea of frozen bits is used to increase the diversity of the genetic material. The caching technique additionally improved the computational preference of the EA.

The results demonstrated usefulness of presented EA approach with new best solutions for three of the standard ORLIB AP instances. Future work would be directed towards parallelization of the EA and its hybridization with exact methods, and also on solving other similar hub location problems.

Acknowledgements. This research was partially supported by Serbian Ministry of Science under the grant no. 144007. The author is grateful to Jozef Kratica for his useful suggestions and comments.

References

1. Ahuja, R., Magnanti, T., Orlin, J.: *Network Flows: Theory, Algorithms and Applications*. Prentice-Hall, New York (1993)
2. Beasley, J.E.: Obtaining Test Problems via Internet. *Journal of Global Optimization* 8, 429–433 (1996), <http://mscmga.ms.ic.ac.uk/jeb/orlib/info.html>
3. Boland, N., Krishnamoorthy, M., Ernst, A.T., Ebery, J.: Preprocessing and Cutting for Multiple Allocation Hub Location Problems. *European Journal of Operational Research* 155, 638–653 (2004)
4. Campbell, J.F.: Hub Location and the p-hub Median Problem. *Operations Research* 44(6), 923–935 (1996)
5. Campbell, J.F., Ernst, A., Krishnamoorthy, M.: Hub Location Problems. In: Hamacher, H., Drezner, Z. (eds.) *Location Theory: Applications and Theory*, pp. 373–407. Springer, Heidelberg (2002)
6. Ernst, A.T., Krishnamoorthy, M.: Exact and Heuristic Algorithms for the Uncapacitated Multiple Allocation p-hub Median Problem. *European Journal of Operational Research* 104, 100–112 (1998)
7. Filipović, V.: Proposition for Improvement Tournament Selection Operator in Genetic Algorithms (in Serbian). Master's thesis, Faculty of Mathematics, University of Belgrade, Serbia (1998)
8. Kang-Ting, M., Ching-Jung, T.: An Ant Colony Optimization Algorithm for Solving the Uncapacitated Multiple Allocation P-Hub Median Problem. In: *Proceedings of The 9th Asia Pasific Industrial Engineering & Management Systems Conference* (2008)
9. Kratica, J.: Improving Performances of the Genetic Algorithm by Caching. *Computers and Artificial Intelligence* 18(3), 271–283 (1999)
10. Reeves, R.C., Rowe, E.J.: *Genetic Algorithms: Principles and Perspectives: A Guide to GA Theory*. Kluwer Academic Publishers, Norwell (2002)
11. Stanimirović, Z.: An Efficient Genetic Algorithm for the Uncapacitated Multiple Allocation p-hub Median Problem. *Control and Cybernetics* 37(3), 669–692 (2007)
12. Stanimirović, Z., Kratica, J., Dugošija, D.: Genetic algorithms for solving the discrete ordered median problem. *European Journal of Operational Research* 182(3), 983–1001 (2007)

Part III

**Soft Computing for Bioinformatics,
Knowledge Management, and Data Mining**

Prediction of the hERG Potassium Channel Inhibition Potential with Use of the Artificial Neural Networks

Sebastian Polak, Barbara Wiśniowska, Malidi Ahamadi,
and Aleksander Mendyk

Abstract. The procedure of drug development is complex and time consuming. Informally this procedure is divided into strongly inter-dependent phases and regulated by the regulatory bodies like FDA (Food and Drug Administration). They provide general guidelines for the pharmaceutical industry. Recently more and more effort has been invested in the early toxicity assessment. One of the possible and potentially dangerous effect is a triggered by drugs acquired long QT syndrome (LQTS) which can lead to the fatal ventricular arrhythmia. In the last decades LQTS was responsible for the withdrawal of several drugs from the market. In most drugs known causing TdP (Torsade de Pointes) the QT segment prolongation in the ECG (electrocardiography) results from inhibition of fast potassium channel (encoded as hERG). Therefore early prediction of the hERG channel interaction potential has become a major pharmacological safety concern for the substances being drug candidates. The objective of this research is to develop a reliable empirical model for the potassium channel inhibition prediction, based on the previously published and transparent database. The input data consisted of parameters describing chemical structure and physico-chemical parameters of the substances. Artificial neural networks were chosen as the algorithms for the models development. Classifiers were built on the training set containing 447 records. Two test modes were applied to the model performance assessment: standard

Sebastian Polak · Barbara Wiśniowska
Unit of Pharmacoepidemiology and Pharmacoeconomics Faculty of Pharmacy,
Collegium Medicum, Jagiellonian University, Cracow, Poland
Medyczna 9 Str. 30-688 Kraków Poland
Tel.: (+) 48 12 62 05 517
e-mail: spolak@cm-uj.krakow.pl

Malidi Ahamadi
Department of Pharmaceutical Technology and Biopharmaceutics Faculty of Pharmacy,
Collegium Medicum, Jagiellonian University, Cracow, Poland
Medyczna 9 Str. 30-688 Kraków Poland

Aleksander Mendyk
Department of Applied Mathematics, University of Leeds, Leeds, LS2 9JT

10-fold cross validation procedure and validation based on the external test set of 45 records. The performance of the best model estimated in 10-fold CV was 76% (78% for positive and 74% for negative output respectively). Best obtained model properly predicted 89% instances from the external validation set.

1 Introduction

The procedure of drug development is complex and time consuming. Informally this procedure is divided into strongly inter-dependent phases and regulated by the regulatory bodies like FDA (Food and Drug Administration) and other independent institutions. They provide general guidelines for the pharmaceutical industry with suggestions for almost all development stages. As the patients' safety is the main concern more and more effort is invested on the early toxicity prediction. Analysis of the clinical data from the last 20 years proved that the drugs withdrawn from the market were caused mainly by the unexpected occurrence of the cardiotoxic effects in patients [7],[18][17]. Their most dangerous clinical symptom is arrhythmia caused by the QT interval increase called Torsade de Pointes which can be clearly observed in the ECG. Such cases resulted in increasing interest in early assessing the cardiotoxic potential of drugs which are currently under development. Nowadays, the assessment of the torsadogenic potency of a new chemical entity is a crucial issue during lead optimization and drug development process and is required by the regulatory agencies during the registration process [1],[6].

2 hERG Channel Inhibition Prediction Model

Human heart action and its physiological function is the result of cardiomyocytes electrophysiological activity which is a consequence of a subtle balance between inward and outward ion currents. Any disturbances in this fine equilibrium can lead to the unwanted cardiac events which can have major consequences including sudden cardiac death. The QT interval in ECG which is measured from the beginning of the QRS complex to the end of the T wave reflects the ventricular depolarization (QRS) and repolarisation (T wave). Drug-induced QT interval prolongation and the appearance of ventricular arrhythmia called Torsade de Pointes (TdP) were recognized as the most important risk factors for patients treated with use of the chosen substances. In most cases TdP is a self-limiting phenomenon but occasionally it may degenerate into potentially fatal ventricular fibrillation. Drugs from various pharmacological groups have demonstrated the potency of inhibiting potassium ions flow which can even lead to the life-threatening situations.

So far hERG potassium channel is the only one proven molecular target for drugs associated with QT prolongation and TdP risk [4]. For this reason it is recognised as a surrogate marker for the proarrhythmic properties of chemicals, although in vitro hERG channel blockade is not synonymous with actual TdP risk in

the clinical settings. Adequate cardiac risk evaluation needs complex and integrative assessment of hERG data and combination of the results from other non-clinical and clinical studies. Nevertheless, as most cases of prolonged duration of cardiac action potential related to drug exposure can be traced to one specific mechanism, namely the blockade of potassium current in the heart (encoded as IKr), at the early stages of drug development, the concentration of the tested substance producing half-maximal block of the hERG potassium current in the in vitro settings (IC₅₀) is a surrogate marker for the pro-arrhythmic properties, and offers rough estimation for human in vivo safety. hERG channel activity assessment is one of the basis for go-no-go decisions in term of further drug development and can provide valuable information. It is reason for the sets of in vitro methods development which enable channel inhibition assessment and common usage in the industry laboratories. Among these methods, electrophysiological assays, especially the Patch Clamp technique in the whole cell mode [11],[21], are considered as a 'gold standard', however there are no precise regulations on experimental method, model or settings selection or standardized protocols for hERG current measurements [14],[20]. Recommendations of the regulatory authorities i.e. ICH S7B define, among others, ion channel studies with the use of either cardiac cells with native hERG channels or miscellaneous heterologous expression systems. There are many non-cardiac cells which can be transfected with hERG cDNA and express hERG potassium channel protein efficiently, nevertheless some of them are exploited most frequently. Cell lines mostly utilized as an expression system for hERG in studies aimed to predict blocking potency for native IKr include HEK 293 cells (human embryonic kidney), CHO cells (Chinese hamster ovary), or XO cells (*Xenopus laevis* oocytes). Each of them has their own, unique characteristic offered for the experimenters and their short descriptions were given in the previous publication [12]. As it could be seen during the literature data analysis, the most significant differences of IC₅₀ values for the same substance can be found when using distinct cell lines and different temperatures as they are highly dependent on the experimental settings. It is one of the reasons for developing inter-system and inter-condition factors which were also used during the data pre-processing [19][6].

Most of the previously published in silico models for the hERG channel inhibition prediction were developed either on the relatively small sets of data, derived based on the various in vitro models or large but unpublished databases. Another one issue which ought to be raised is inconsistency of the experimental data and problem caused by that fact in term of building classification models. Some researchers decide not to use problematic data points from the native dataset and to develop models which are not fully suitable for the practical usage at the early drug development stage in term of the fidelity. The aim of the research was to develop a reliable model for the potassium channel inhibition prediction, based on the previously published and transparent database with use of the scaling factors for the data pre-processing [12],[19] without excluding confusing data points to make the model usable for the daily screening needs.

3 Computational Methods

3.1 Algorithm

Artificial neural networks (ANNs) are well-established empirical modeling tools. Their main advantage is that they are able to learn automatically from available data providing predictions (generalization) as well as some insights into the hidden relationships (data-mining). In this work classical multi-layer perceptrons (MLP) and neuro-fuzzy systems of Mamdani MISO (multiple input single output) type were applied. The task for the ANNs to solve was binary classification of drug safety based on the drug description provided at the input. There were 1034 inputs describing each drug compound and one output providing safety assessment by a simple binary decision.

The ANNs optimal architecture search was carried out by comparison of generalization error of various architectures-candidates in the 10-fold cross-validation procedure. There were several types of MLPs architectures trained and tested: 1 to 4 hidden layers with up to 80 nodes in each; activation functions like hyperbolic tangent, Heaviside, linear and fsr (a specific combination of two logarithmic functions); adjacent layers fully interconnected.

All the ANNs were trained by backpropagation algorithm with momentum and jog-of-weights modifications. The latter is the stochastic search algorithm designed to add noise to the weights after the specified number of iterations had passed without learning improvement – the aforementioned patience criterion was set to 50 000. The number of training iterations was varied from 50 000 to 10 000 000. The epoch size was 1. Training of ANNs involved random data records presentation with or without adding a noise to the data. The data was scaled linearly from 0.2 to 0.8 or from -0.8 to 0.8. For neuro-fuzzy systems the M-SLIDE defuzzification method was applied and initialization of ANN rules based directly on the random data records chosen. Root mean squared error was chosen as the error measure. There were around 5 000 neural models. All ANN architectures were encoded in the same way (number of neurons in subsequent hidden layers_activation function_size of the input vector_problem description_dataset presentation) ie. 3_2_sigma_1034_in_CHEK_rand what means that there are two hidden layers (3 and 2 neurons respectively), with sigmoidal activation function, the total number of inputs is 1034 and the records from dataset were presented in a random way.

3.2 Software

An own written ANN simulator *Nets2009* was used to perform the model development. The simulator was written in Pascal (Lazarus) [9] and compiled either with or without GUI in order to provide a tool for models preparation and analysis (GUI) as well as for direct training of ANNs (console mode). In order to handle computational jobs there was a software developed (*dissolve*) based on Linux *bash*

scripts and *cron* daemon, which automatically managed deployment and retrieval of computational jobs in a grid-like structure. The communication between workstations was maintained based on the SSH protocol.

4 Data Set and Data Preprocessing

The original data set containing the data collected from available scientific literature sources was previously described. As the data quality was our large concern every single report and data record was carefully checked to meet the inclusion criteria [12]. The final dataset consisted of 447 hERG IC₅₀ values and is freely available as a supplementary material (<http://www.tox-portal.net>) apart of the whole in vitro cardiotoxicity database available at the same place. The dataset contains information for 173 different molecules from both groups - with proven hERG blocking activity and without it. All data were obtained experimentally during in vitro assays performed on three different cell-line types HEK, CHO or XO. IC₅₀ values, expressed in micromolar units (μM), were obtained with use of the whole-cell patch clamp method for HEK and CHO and 2-electrode voltage clamp for XO by fitting experimental data to the Hill equation. We note here that the experimental settings, especially in vitro model and the temperature maintained during the assays can affect the results considerably. Internally derived inter-system and inter-temperature extrapolation factors were used in order to standardize the data [12],[19],[6].

4.1 Test Sets

There were two modes of the developed models performance verification and accordingly two types of the test sets. Initially model was validated during the standard 10-fold cross-validation procedure (10-CV) and then the final testing was done based on their classification efficacy on the external test records. The algorithm used for the learn-test pair generation solves multiple knapsack problem assigning to each test set at most 10% of the training set records separating compounds between learning and testing sets. The algorithm run resulted in test sets as presented in Table 1.

In order to test model generalization, an external set of data was collected. Validation set comprised 45 records which were not previously used during the model development. This includes 23 new records for 16 different compounds already present in the main database. As some of the validation records related to the compounds that were already present during the model development the validation set underwent further division to another two validation subsets encoded as "new" and "old". The "new" one contains 22 records for the 16 different compounds with the chemical structure completely unknown to the model whereas the "old" enclosed exclusively records build based on the compounds used during model construction.

Table 1 Test sets used during 10-CV procedure

Test set	No. of records	No. of different compounds
1	51	3
2	50	4
3	50	5
4	44	6
5	44	9
6	44	14
7	45	21
8	44	35
9	44	44
10	33	30

4.2 Inputs

Input vector consisted of two groups of descriptors. The first one consists of seven represent physicochemical properties of the compounds (MW, RBN, nHDon, nHAcc, nHBonds, TPSA, ALOGP) which were selected arbitrary from those generated by DRAGON software [15]. The selection was based on the assumption of practicality – chosen descriptors are commonly used at this stage of the drug development and are routinely measured and/or calculated in the pharmaceutical laboratories. Two dimensional (2D) structures were represented by molecular fingerprint (substructure or hashed fingerprint) – a binary string of defined number of bits. The core of the computational method used during the study was derived from the Chemical Descriptors Library project [8] and necessary changes and improvements in the code were done. The fingerprint depth - a maximum number of bonds separating two atoms – was set to 7 and the defined fingerprint length was 1024 (every single chemical compound was described by 1024 features). The rest of the input vector elements were parameters describing experimental procedure (depolarization pulse, voltage of the hERG current measurement and type of protocol applied to elicit hERG current with or without model and temperature).

4.3 Outputs

Binary hERG classification models described in the literature are based on different cut-off values discriminating safe and unsafe compounds. Proposed thresholds range from 0.13 μM [5] to 40 μM [1]. Herein all records were assigned into two classes (hERG blockers or non-blockers), with a threshold set at an IC₅₀ value of 1 based on the literature analysis - 1 μM is considered as the un-safety

[13][10][14][16][17]. Based on the threshold value for blocker/non-blocker of 1 μM 250 compounds were classified as active and 197 compounds as inactive.

5 Results and Discussion

The best model based on 1 μM threshold was obtained using BP ANN with 3 and 2 cells in hidden layers respectively. The performance of the best model estimated in standard 10-fold cross-validation procedure was 76% with 78% of safe and 74% of unsafe predicted correctly. The algorithm was able to predict 341 training instances, whereas 106 of them were classified incorrectly. However, due to specific character of the training set it is impossible to obtain 100% of accuracy. For 22 compounds there are contradictory records (123 instances) in the data set, that carry very similar input information with the opposite outputs, thus as the true class of the compound is unknown it is unfeasible for the model to predict them properly. The results of the 10-CV and external validation procedure for the best architectures are presented in table 2 and 3.

Table 2 Results of best obtained architectures - 10CV

No. Architecture	iterations	ALL %	1 %	0 %	PPV	NPV
1. 3_2_sigma_1034_in_cHEK_rand	10 mln	76.24	78	74.11	0.79	0.73
2. 3_2_tanh_1034_in_cHEK_orig	1 mln	75.53	76.4	74.62	0.79	0.71
3. 3_2_tanh_1034_in_cHEK_rand	10 mln	74.54	71.6	78.68	0.81	0.69
4. 60_20_8_fsr_1034_in_cHEK_orig	500 k	74.00	71.6	76.65	0.80	0.68
5. 60_20_8_tanh_1034_in_cHEK_orig	1 mln	73.86	73.2	74.11	0.78	0.69

ALL % – total classification rate; 1 (%) – classification rate of records with 1 value; 0 (%) – classification rate of records with 0 value.

Table 3 Results of external validation procedure

No	Architecture	validation set	ALL %	1 %	0 %
		ALL	84	88	80
1.	3_2_sigma_1034_in_cHEK_rand	NEW	82	71	77
		OLD	87	90	90
		ALL	84	75	92
2.	3_2_tanh_1034_in_cHEK_orig	NEW	86	71	93
		OLD	83	77	90
		ALL	87	80	92
3.	3_2_tanh_1034_in_cHEK_rand	NEW	86	86	87
		OLD	87	77	100

Table 3 (continued)

4.	60_20_8_fsr_1034_in_cHEK_orig	NEW	82	57	93
		OLD	87	85	90
		ALL	89	85	92
5.	60_20_8_tanh_1034_in_cHEK_orig	NEW	91	86	93
		OLD	87	85	90

ALL % – total classification rate; 1 (%) – classification rate of records with 1 value; 0 (%) – classification rate of records with 0 value; ALL – total validation set; NEW – only new compounds; OLD – compounds present in training set.

6 Conclusions

As mentioned before, experimental data can differ considerably since they are highly dependent on experimental settings. IC50 values disparities can entail a problem of unequivocal compounds discrimination and substances with such properties may possibly be classified into different classes. These discrepancies were solved to some extent with use of recently proposed extrapolation factors. Nevertheless, there still remain some other issues, i.e. inter-laboratory variability of the results. In our view anyway there is no rationale for arbitrary assigning a compound to one class or another if experimental results are ambiguous but assume some level of uncertainty which has to be taken under consideration during model usage.

Acknowledgments. We would like to acknowledge Mr Wojciech Wiśniowski for his generous help and providing C code expertise.

References

- [1] Anonymus. Guidance for Industry S7B Nonclinical Evaluation of the Potential for Delayed Ventricular Repolarization (QT Interval Prolongation) by Human Pharmaceuticals (2005)
- [2] Aronov, A.M., Goldman, B.B.: A model for identifying HERG K⁺ channel blockers. *Bioorg. Med. Chem.* 12(9), 2307–2315 (2004)
- [3] Bains, W., Basman, A., White, C.: HERG binding specificity and binding site structure: evidence from a fragment-based evolutionary computing SAR study. *Progr. Biophys. Mol. Biol.* 86, 205–233 (2004)
- [4] Brown, A.M.: hERG block, QT liability and sudden death. In: Derek, J., Goode, C., Goode, J. (eds.) *The hERG cardiac potassium channel: structure, function and long QT syndrome*. ed., Novartis Foundation, John Wiley & Sons, Chichester (2005)
- [5] Buyck, C., Tollenaere, J., Engels, M., et al.: *The 14th European Symposium on QSARs*, Bournemouth (UK), September 8–13 (2002)

- [6] Darpö, B., Nebout, T., Sager, P.: Clinical evaluation of QT/QTc prolongation and proarrhythmic potential for nonantiarrhythmic drugs: The International Conference on Harmonization of Technical Requirements for Registration of Pharmaceuticals for Human Use E14 guideline. *J. Clin. Pharmacol.* 46(5), 498–507 (2006)
- [7] Fung, M., Thornton, A., Mybeck, K., et al.: Evaluation of the characteristics of safety withdrawal of prescription drugs from worldwide pharmaceutical markets – 1960 to 1999. *Drug Inform J.* 35, 293–317 (2001)
- [8] <http://cdelib.sourceforge.net/doc/fingerprints.html>
- [9] <http://www.lazarus.freepascal.org/>
- [10] Keserü, G.M.: Prediction of hERG potassium channel affinity by traditional and hologram qSAR methods. *Bioorg. Med. Chem. Lett.* 13(16), 2773–2775 (2003)
- [11] Neher, E., Sakmann, B.: The patch clamp technique. *Sci. Am.* 266(3), 44–51 (1992)
- [12] Polak, S., Wiśniowska, B., Brandys, J.: Collation, assessment and analysis of literature in vitro data on hERG receptor blocking potency for subsequent modeling of drugs cardiotoxic properties. *J. Appl. Tox.* 29(3), 183–206 (2009)
- [13] Roche, O., Trube, G., Zuegge, J., Pflimlin, P., Alanine, A., Schneider, G.: A virtual screening method for prediction of the hERG potassium channel liability of compound libraries. *Chem. Bio. Chem.* 3(5), 455–459 (2002)
- [14] Sahah, R.R.: Drug-induced QT interval prolongation – regulatory guidance and perspectives on hERG channel studies. In: Derek, J., Goode, C., Goode, J. (eds.) *The hERG cardiac potassium channel: structure, function and long QT syndrome*. ed., Novartis Foundation. John Wiley & Sons, Chichester (2005)
- [15] Talete srl, DRGON for Windows (Software for Molecular Descriptor Calculations). Version 5.5 (2007), <http://www.talete.mi.it/>
- [16] Thai, K.M., Ecker, G.F.: A binary QSAR model for classification of hERG potassium channel blockers. *Bioorg. Med. Chem.* 16, 4107–4119 (2008)
- [17] Tobita, M., Nishikawa, T., Nagashima, R.: A discriminant model constructed by the support vector machine method for hERG potassium channel inhibitors. *Bioorg. Med. Chem. Lett.* 15(11), 2886–2890 (2005)
- [18] U.S. Food and Drug Administration. CDER, Report to the Nation: Improving Public Health Through Human Drugs. Rockville, Maryland (2005)
- [19] Wiśniowska, B., Polak, S.: hERG in vitro interchange factors – development and verification. *Tox. Mech. Met.* 19(4), 278–284 (2009)
- [20] Witchel, H., Milnes, J., Mitcheson, J., et al.: Troubleshooting problems with in vitro screening of drugs for QT interval prolongation using hERG K⁺ channels expressed in mammalian cell lines and *Xenopus* oocytes. *J. Pharmacol. Toxicol. Methods* 48(2), 65–80 (2002)
- [21] Wood, C., Williams, C., Waldron, G.J.: Patch clamping by numbers. *Drug Discov. Today* 9, 434–441 (2004)

Michigan Style Fuzzy Classification for Gene Expression Analysis

Gerald Schaefer and Tomoharu Nakashima

Abstract. Interest in microarray studies and gene expression analysis is growing as they are likely to provide promising avenues towards the understanding of fundamental questions in biology and medicine. In this paper we employ a hybrid fuzzy rule-based classification system for effective analysis of gene expression data. Our classifier consists of a set of fuzzy if-then rules that allows for accurate non-linear classification of input patterns. A small number of fuzzy if-then rules are selected through means of a genetic algorithm in order to provide a compact classifier for gene expression analysis. Experimental results on various well-known gene expression datasets confirm the efficacy of the presented approach.

1 Introduction

Microarray expression studies measure, through a hybridisation process, the levels of genes expressed in biological samples. Knowledge gained from these studies is deemed increasingly important due to its potential of contributing to the understanding of fundamental questions in biology and clinical medicine. Microarray experiments can monitor each gene several times under varying conditions or analyse the genes in a single environment but in different types of tissue. In this paper we focus on the latter where one important aspect is the classification of the recorded samples. This can be used to categorise different types of cancerous tissues as in (Golub et al., 1999) where different types of leukemia are identified, or to distinguish cancerous tissue from normal tissue as done by (Alon et al. (1999)) who analysed tumor and normal colon tissues.

Gerald Schaefer

Department of Computer Science, Loughborough University, U.K.

Tomoharu Nakashima

Department of Computer Science and Intelligent Systems,
Osaka Prefecture University, Japan

In this paper, we apply a hybrid GA-fuzzy classification scheme to analyse microarray expression data. Our classifier consists of a set of fuzzy if-then rules that allow for accurate non-linear classification of input patterns. A small number of fuzzy if-then rules are selected through means of a genetic algorithm to arrive at a compact yet effective rule base. Experimental results on several gene expression datasets show that this approach affords classification performance comparable with the classifier used in our earlier work (Schaefer et al., 2007) where a much larger rule base was necessary.

2 Fuzzy Rule-Based Classification

Pattern classification is typically a supervised process where, based on set of training samples with known classifications, a classifier is derived that performs automatic assignment of unseen data to pre-defined classes. Let us assume that our pattern classification problem is an n -dimensional problem with C classes and m given training patterns $\mathbf{x}_p = (x_{p1}, x_{p2}, \dots, x_{pn})$, $p = 1, 2, \dots, m$. Without loss of generality, we assume each attribute of the given training patterns to be normalised into the unit interval $[0, 1]$; that is, the pattern space is an n -dimensional unit hypercube $[0, 1]^n$. We use fuzzy if-then rules of the following type as a base of our fuzzy rule-based classifier

$$\begin{aligned} \text{Rule } R_j: & \text{ If } x_1 \text{ is } A_{j1} \text{ and } \dots \text{ and } x_n \text{ is } A_{jn} \\ & \text{ then Class } C_j \text{ with } CF_j, \quad j = 1, 2, \dots, N, \end{aligned} \quad (1)$$

where R_j is the label of the j -th fuzzy if-then rule, A_{j1}, \dots, A_{jn} are antecedent fuzzy sets on the unit interval $[0, 1]$, C_j is the consequent class (i.e., one of the C given classes), and CF_j is the grade of certainty of the fuzzy if-then rule R_j .

Our fuzzy rule-based classification system consists of N rules each of which has a form as in Equation (1). There are two steps in the generation of fuzzy if-then rules: specification of the antecedent part and determination of consequent class C_j and the grade of certainty CF_j . The antecedent part is specified manually (we use triangular fuzzy sets). Then, the consequent part (i.e., consequent class and the grade of certainty) is determined from given training patterns. In (Ishibuchi and Nakashima, 2001) it is shown that the use of the grade of certainty in fuzzy if-then rules allows us to generate comprehensible fuzzy rule-based classification systems with high classification performance. Due to space restrictions, we refer the reader to (Schaefer et al., 2007) for details on how to derive the class and grade of certainty for each rule, and how a single winner rule is selected to classify previously unseen data.

3 Hybrid Fuzzy Classification

While the basic fuzzy rule-based system provides a reliable and accurate classifier, it suffers - as do many other approaches - from the curse of dimensionality. In the case of our fuzzy classifier, the number of generated rules increases exponentially with the number of attributes involved and with the number of partitions used for each attribute. We are therefore interested in arriving at a more compact classifier that affords the same classification performance while not suffering from the problems.

In our previous work in (Schaefer et al., 2007), we applied a rule splitting step to deal with the problem of dimensionality. By limiting the number of attributes in each rule to 2, a much smaller rule base was developed. In this paper, we take a different approach to arrive at an even more compact rule base by developing a hybrid fuzzy classification system through the application of a genetic algorithm (GA). The fuzzy if-then rules used do not change and are still of the same form as the one given in Equation (II), i.e., they contain a number of fuzzy attributes and a consequent class together with a grade of certainty. Our approach of using GAs to generate a fuzzy rule-based classification system is a Michigan style algorithm (Ishibuchi and Nakashima, 1999) which represents each rule by a string and handles it as an individual in the population of the GA. A population consists of a pre-specified number of rules. Because the consequent class can be derived from the given training patterns, they are not used in the coding of each fuzzy rule (i.e., they are not included in a string). Each rule is hence represented by a string using its antecedent fuzzy sets.

3.1 Genetic Operations

First, the algorithm randomly generates a pre-specified number N_{rule} of rules as an initial population (in our experiments we set $N_{\text{rule}} = 20$). Next the fitness value of each fuzzy rule in the current population is evaluated. Let S be the set of rules in the current population. The evaluation of each rule is performed by classifying all training patterns by the rule set S using a single winner-based method. The winning rule receives a unit reward when it correctly classifies a training pattern. After all the given training patterns are classified by the rule set S , the fitness value $fitness(R_q)$ of each rule R_q in S is calculated as

$$fitness(R_q) = NCP(R_q), \quad (2)$$

where $NCP(R_q)$ is the number of correctly classified training patterns by R_q . It should be noted that the following relation holds between the classification performance $NCP(R_q)$ of each rule R_q and the classification performance $NCP(S)$ of the rule set S used in the fitness function:

$$NCP(S) = \sum_{R_q \in S} NCP(R_q). \quad (3)$$

The algorithm is implemented so that only a single copy is selected as a winner rule when multiple copies of the same rule are included in the rule set S . In GA optimisation problems, multiple copies of the same string usually have the same fitness value. This often leads to undesired early convergence of the current population to a single solution. In our algorithm, only a single copy can have a positive fitness value and the other copies have zero fitness which prevents the current population from being dominated by many copies of a single or few rules.

New rules are generated from the rules in the current population using genetic operations. As parent strings, two fuzzy if-then rules are selected from the current population and binary tournament selection with replacement is applied. That is, two rules are randomly selected from the current population and the better rule with the higher fitness value is chosen as a parent string. A pair of parent strings is chosen by iterating this procedure twice.

From the selected pair of parent strings, two new strings are generated by a crossover operation. We use a uniform crossover operator where the crossover points are randomly chosen for each pair of parent strings. The crossover operator is applied to each pair of parent strings with a pre-specified crossover probability p_c . After new strings are generated, each symbol of the generated strings is randomly replaced with a different symbol by a mutation operator with a pre-specified mutation probability p_m (we use the same mutation probability for every position of a string). Selection, crossover, and mutation are iterated until a pre-specified number N_{replace} of new strings are generated.

Finally, the N_{replace} strings with the smallest fitness values in the current population are removed, and the newly generated N_{replace} strings added to form a new population. Because the number of removed strings is the same as the number of added strings, every population consists of the same number of strings. That is, every rule set has the same number of rules. This generation update can be viewed as an elitist strategy where the number of elite strings is $(N_{\text{rule}} - N_{\text{replace}})$.

The above procedures are applied to the new population again. The generation update is iterated until a pre-specified stopping condition is satisfied. In our experiments we use the total number of iterations (i.e., the total number of generation updates) as stopping condition.

3.2 Algorithm Summary

To summarise, our hybrid fuzzy rule-based classifier works as follows:

Step 1: *Parameter Specification*: Specify the number of rules N_{rule} , the number of replaced rules N_{replace} , the crossover probability p_c , the mutation probability p_m , and the stopping condition.

- Step 2: *Initialisation*: Randomly generate N_{rule} rules (i.e., N_{rule} strings of length n) as an initial population.
- Step 3: *Genetic Operations*. Calculate the fitness value of each rule in the current population. Generate N_{replace} rules using selection, crossover, and mutation from existing rules in the current population.
- Step 4: *Generation Update (Elitist Strategy)*: Remove the worst N_{replace} rules from the current population and add the newly generated N_{replace} rules to the current population.
- Step 5: *Termination Test*: If the stopping condition is not satisfied, return to Step 3. Otherwise terminate the execution of the algorithm.

During the execution of the algorithm, we monitor the classification rate of the current population on the given training patterns. The rule set (i.e., population) with the highest classification rate is chosen as the final solution.

3.3 Increasing the Classification Performance

Various strategies were pursued to achieve improved classification performance. In particular, we

- introduce a concept of “*don't care*” attributes for antecedent fuzzy sets to allow rules to cover a wider input pattern space.
- use training patterns for specifying antecedent fuzzy sets.
- use antecedent fuzzy sets specified from training patterns also for the update of the GA population.
- introduce a penalty term with respect to the number of misclassified training patterns.

Due to space restrictions we refer the reader to (Schaefer and Nakashima, to appear) for a detailed discussion of these strategies and their application to gene expression classification.

4 Experimental Results

We evaluated our proposed method on three gene expression datasets that are commonly used in the literature, namely the Colon (Alon et al., 1999), Leukemia (Golub et al., 1999), and Lymphoma (Alizadeh et al., 2000) data sets. Although these represent only 2-class or 3-class problems, due to the large number of genes involved, any rule-based classification system would consist of a very large number of rules and hence represent a fairly complex process. Also, not all genes are equally important for the classification task at hand. We therefore sort the significance of genes according to the BSS/WSS (the ratio of between group to within group sum of squares) criterion used in (Dudoit et al., 2002) and consider only the top 50 respectively 100 genes as input for our classification problem.

We perform standard leave-one-out cross-validation where classifier training is performed on all available data except for the sample to be classified and this process is performed for all samples¹. Fuzzy rule-based classifiers as in (Schaefer et al., 2007) and hybrid fuzzy classifiers (comprising 20 rules) as proposed in this paper, based on partition sizes L of 2 and 5 partitions for each gene, were constructed. For comparison, we also implemented a simple nearest neighbour classifier which searches through the complete training data to identify the sample which is closest to a given test input and assigns the identified sample's class. The results on the three datasets are given in Table 1 and are reported in terms of classification accuracy, i.e. the percentage of correctly classified samples.

Table 1 Classification results on the three datasets

n	classifier	Colon	Leukemia	Lymphoma
50	fuzzy $L = 2$	80.65	91.67	95.74
	fuzzy $L = 3$	85.48	94.44	97.87
	fuzzy $L = 4$	83.87	93.06	100
	fuzzy $L = 5$	77.42	91.67	93.62
	hybrid fuzzy $L = 2$	80.11	96.29	97.88
	hybrid fuzzy $L = 3$	83.87	96.29	92.20
	hybrid fuzzy $L = 4$	79.03	93.98	90.07
	hybrid fuzzy $L = 5$	80.64	91.20	91.49
	nearest neighbour	79.03	97.22	95.74
	100	fuzzy $L = 2$	70.97	87.50
fuzzy $L = 3$		82.26	98.61	93.62
fuzzy $L = 4$		80.65	95.83	93.62
fuzzy $L = 5$		74.19	93.06	82.98
hybrid fuzzy $L = 2$		78.01	94.44	93.61
hybrid fuzzy $L = 3$		80.11	93.52	89.65
hybrid fuzzy $L = 4$		76.88	88.74	80.14
hybrid fuzzy $L = 5$		77.42	87.50	73.76
nearest neighbour		83.87	97.22	100

Looking at the results for the Colon dataset, for the case of 50 selected features the fuzzy classifier with 3 partitions performs best with a classification accuracy of 85.48% which corresponds to 9 incorrectly classified cases, while nearest neighbour classification produces 13 errors. However, when selecting the 100 top genes the nearest neighbour classifier performs slightly better than the fuzzy system. It is interesting to compare the performance of the fuzzy rule-based classifier when using different numbers of partitions for each attribute. It can be seen that on this dataset the best performance is

¹ It should be noted that the top 50 respectively 100 genes are selected based solely on the training set.

achieved when using 3 partitions (although on training data alone more partitions afford better performance). In particular, it can be observed that the case with $L = 2$ as used in the work of Vinterbo *et al.* (Vinterbo et al., 2005) produces the worst results and hence confirms that increasing the number of fuzzy intervals improves the classification performance (Schaefer et al., 2007). However, it can also be seen that applying too many partitions can decrease classification performance as is apparent in the case of $L = 5$. For the hybrid fuzzy classifier we ran the experiment 10 times with different, random, initial populations and list the average of the 3 best runs. We can see the the hybrid fuzzy approach performs only slightly worse than the full fuzzy rule-based system which, considering that classification is performed based only on 20 selected rules, proves the potential of this method.

Turning our attention to the results on the Leukemia dataset, we see a similar picture. Again, the worst performing fuzzy classifier is that which uses only two partitions per gene while the best performing one as assessed by leave-one-out cross validation is the case of $L = 3$. Nearest neighbour classification performs well again confirming previous observations that despite its simplicity, nearest neighbour classifiers are well suited for gene expression classification (Dudoit et al., 2002). The best classification results are achieved by the fuzzy classifier with $L = 3$ for the case of 100 selected genes with a classification accuracy of 98.61% and the nearest neighbour classifier with 97.22% for 50 selected genes. For the case of 50 features, the hybrid fuzzy classifier outperforms the conventional fuzzy classification system in most cases while for classification based on 100 features it is more accurate only for the case of $L = 2$.

Finally, for the Lymphoma dataset, perfect classification is achieved by the fuzzy classifier with $L = 4$ for 50 selected genes and by nearest neighbour classification based on 100 genes. The hybrid fuzzy approach performs slightly worse in most cases but is significantly worse for $L = 4$ and $L = 5$ and 100 features. The reason for this performance is probably that during the run, the genetic algorithm was unable to sufficiently explore all of the search space within the number of iterations allowed.

5 Conclusions

In this paper we have applied a hybrid GA-fuzzy classifier to the problem of classifying gene expression data. The presented classifier consists of a set of fuzzy if-then rules that allows for accurate non-linear classification of input patterns. A compact yet effective rule base is constructed through the application of a genetic algorithm to select useful rules. Experimental results have shown this classifier to work well for gene expression analysis. Also, the classification performance of the hybrid method was shown to be comparable to that of the full classifier.

Acknowledgements. The support of the Sasakawa Foundation and the Daiwa Foundation for this work is greatly acknowledged.

References

- Alizadeh, A., Eisen, M., Davis, E., Ma, C., Lossos, I., Rosenwald, A., Boldrick, J., Sabet, H., Tran, T., Yu, X., Powell, J., Yang, L., Marti, G., Moore, T., Hudson, J., Lu, L., Lewis, D., Tibshirani, R., Sherlock, G., Chan, W., Greiner, T., Weisenburger, D., Armitage, J., Warnke, R., Levy, R., Wilson, W., Grever, M., Byrd, J., Botstein, D., Brown, P., Staudt, L.: Different types of diffuse large B-cell lymphoma identified by gene expression profiles. *Nature* 403, 503–511 (2000)
- Alon, U., Barkai, N., Notterman, D., Gish, K., Ybarra, S., Mack, D., Levine, A.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natnl. Acad. Sci. USA* 96, 6745–6750 (1999)
- Dudoit, S., Fridlyand, J., Speed, T.: Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association* 97(457), 77–87 (2002)
- Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* 286, 531–537 (1999)
- Ishibuchi, H., Nakashima, T.: Improving the performance of fuzzy classifier systems for pattern classification problems with continuous attributes. *IEEE Trans on Industrial Electronics* 46(6), 1057–1068 (1999)
- Ishibuchi, H., Nakashima, T.: Effect of rule weights in fuzzy rule-based classification systems. *IEEE Trans Fuzzy Systems* 9(4), 506–515 (2001)
- Schaefer, G., Nakashima, T.: Data mining of gene expression data by fuzzy and hybrid fuzzy methods. *IEEE Trans. on Information Technology in Biomedicine* (to appear)
- Schaefer, G., Nakashima, T., Yokota, Y., Ishibuchi, H.: Fuzzy classification of gene expression data. In: *IEEE Int. Conference on Fuzzy Systems*, pp. 1090–1095 (2007)
- Vinterbo, S., Kim, E.Y., Ohno-Machado, L.: Small, fuzzy and interpretable gene expression based classifiers. *Bioinformatics* 21(9), 1964–1970 (2005)

Feature Selection for Bankruptcy Prediction: A Multi-Objective Optimization Approach

Fernando Mendes, João Duarte, Armando Vieira, and António Gaspar-Cunha

Abstract. In this work a Multi-Objective Evolutionary Algorithm (MOEA) was applied for feature selection in the problem of bankruptcy prediction. The aim is to maximize the accuracy of the classifier while keeping the number of features low. A two-objective problem - minimization of the number of features and accuracy maximization - is fully analyzed using two classifiers: Support Vector Machines and Logistic Function. A database containing financial statements of 1200 medium sized private French companies was used. It was shown that MOEA is a very efficient feature selection approach. Furthermore, it can provide very useful information for the decision maker in characterizing the financial health of a company.

1 Introduction

Financial bankruptcy prediction is of high importance for banks, insurance companies, creditors and investors. Different approaches have been used in this type of prediction like linear discriminant analysis and non-linear models. However, these methods have some limitations. Discriminant analysis is limited due to its linearity, restrictive assumptions, for treating financial ratios as independent variables and can only be used with continuous independent variables. In non-linear models the choice of the regression function creates a bias that restricts the outcome, they are very sensitive to exceptions, and most conclusions have an implicit Gaussian distribution on data, which is inappropriate in many cases [1, 2].

More recently other approaches have been applied for bankruptcy classification, such as Artificial Neural Networks (ANN) [3], Evolutionary Algorithms

Fernando Mendes · António Gaspar-Cunha
IPC/I3N - Institute of Polymers and Composites, University of Minho,
Guimarães, Portugal
e-mail: {fmendes, agc}@dep.uminho.pt

João Duarte · Armando Vieira
Department of Physics, Instituto Superior de Engenharia do Porto,
R. S. Tomé, 4200 Porto, Portugal
e-mail: {jmm, asv}@isep.ipp.pt

(EA) and Support Vector Machines (SVM) [4]. ANN, EA and SVM are used as complementary tools to classify credit risk. Some of the studies performed show that ANN outperforms discriminant analysis in bankruptcy prediction [5-7]. These models have shown promising results. However it is generally recognized that further research is needed to achieve higher predictive capabilities, which is the avenue of our research.

Since these methods can handle a large number of variables, and some of them are highly correlated, it is of crucial importance to have a feature selection algorithm to reduce the number of features considered for analysis. In this work a methodology based on MOEA to accomplish simultaneously two objectives: the minimization of the number of features used and the maximization of the accuracy of the classifier used is proposed. The use of MOEA for solving this problem is not new. Merelo et al. [8] applied a MOEA to take into account individually the errors of type I (false positive) and type II (false negative). Hamdani et al. [9] used the NSGA-II [10] algorithm to optimize simultaneously the number of features and the global error obtained by a neural network classifier.

A database containing financial statements of 1200 small and medium size private French companies during 2006 was been used. Each company is characterized by a set of 30 features measuring its most important ratios, from profitability to debt.

This text is organized as follows. In section 2 the problem to solve will be explained in more detail as well the classification methods used here (RG and SVM). Then, in section 3 the MOEA used will be described in more detail. The method proposed will be applied to a case study and the results will be presented and discussed in section 4. Finally, the conclusion will be stated in section 5.

2 Bankruptcy Prediction

Bankruptcy prediction is a problem stated as follows: given a set of financial statements from a company over one or several previous years, predict the changes that it will become distressed over a given period, normally the next year or two years ahead.

The first step consists in “cleaning up” the database in order to create a well balanced and unbiased sample. The dataset must be simplified in order to the problem be more understandable by the decision maker. This can be accomplished by reducing the number of features necessary. On the other hand it must be assured that this reduction does not decrease the performance of the classifier.

The methodology proposed in this work uses the data classifiers to measure the accuracy of the data (or the features used), while a MOEA will be used to determine the best compromise between the refereed two objectives. To test the ability of the method proposed to accomplish these objectives two classifiers will be used: Logistic Regression (LR) and Support Vector Machines (SVM).

LR is a well known generalized linear method, allowing the prediction of a discrete outcome (generally binary, such as success/failure), from a set of variables that may be continuous, discrete, binary, or a mix of any of these [11]. In the present case the LR was trained by Stochastic Gradient Decent, which is able to

estimates the maximum likelihood logistic regression coefficients from sparse input data.

Support Vector Machines (SVMs) are a set of supervised learning methods based on the use of a kernel, which can be applied to classification and regression. In the SVM a hyperplane or set of hyperplanes is (are) constructed in a high-dimensional space. In this case, a good separation is achieved by the hyperplane that has the largest distance to the nearest training datapoints of any class. Thus, the generalization error of the classifier is lower when this margin is larger. SVMs can be seen an extension to nonlinear models of the generalized portrait algorithm developed by Vladimir Vapnik [12].

3 Multi-objective Optimization

MOEAs have been recognized in the last decade as good methods to explore and find an approximation to the Pareto-optimal front for multi-objective optimization problems. This is due to the difficulty of traditional exact methods to solve this type of problems and by their capacity to explore and combine various solutions to find the Pareto front in a single run. A MOEA must provide a homogeneous distribution of the population along the Pareto frontier, together with an improvement of the solutions along successive generations [13, 14]. In this work, the Reduced Pareto Set Genetic Algorithm (RPSGA) is adopted [15, 16], where a clustering technique is applied to reduce the number of solutions on the efficient frontier. Detailed information about this algorithm can be found elsewhere [15, 16].

In the present study the RPSGA algorithm was adapted to deal with the features selection problem. First it was necessary to identify clearly which are the decision variables, and then the RPSGA was to be linked with the classification algorithms in order to obtain the information about the performance of the classifier.

In what concerns the definition of the decision variables, two different possibilities were considered. Initially, a problem where only the features are to be chosen was studied. In this situation the parameters of the classifiers, such as type of training (holdout or k-fold cross validation), learning rate and training fraction, for both LR and SVM, kernel type and other parameter only for SVM, must be previously defined. In a second approach, these parameter where also included as variables to be optimized. The advantage of the latter approach consists in obtaining in a single run the best results. The methodology proposed will be better illustrated with the example presented in the next section.

4 Results and Discussion

4.1 Case Studies

Let us now illustrate the use of above MOEA methodology to solve a feature extraction problem. The problem to be addressed is a classification problem. The aim is to find the minimum number of features needed to obtain the maximum

accuracy of the companies' evaluation. Accuracy is defined as the number of companies correctly predicted divided by the total number of companies in the sample.

Table 1 shows the features definitions considered in the database. Regarding the company evaluation, the classifier output predicts whether the company has survived or gone into bankruptcy.

Table 1 Set of features considered in the problem to solve

Feature	Designation
F1	Number of employees
F2	Capital Employed / Fixed Assets
F3	Financ. Debt / Capital Employed (\%)
F4	Depreciation of Tangible Assets (\%)
F5	Working capital / current assets
F6	Current ratio
F7	Liquidity ratio
F8	Stock Turnover days
F9	Collection period
F10	Credit Period
F11	Turnover per Employee (thousands euros)
F12	Interest / Turnover
F13	Debt Period days
F14	Financial Debt / Equity (\%)
F15	Financial Debt / Cashflow
F16	Cashflow / Turnover (\%)
F17	Working Capital / Turnover (days)
F18	Net Current Assets/Turnover (days)
F19	Working Capital Needs / Turnover (\%)
F20	Export (\%)
F21	Value added per employee
F22	Total Assets / Turnover
F23	Operating Profit Margin (\%)
F24	Net Profit Margin (\%)
F25	Added Value Margin (\%)
F26	Part of Employees (\%)
F27	Return on Capital Employed (\%)
F28	Return on Total Assets (\%)
F29	EBIT Margin (\%)
F30	EBITDA Margin (\%)

In the case of LR, several runs were performed using the gradient descent method and various combinations of the training method (holdout and 10-fold validation), the learning rate (0.001, 0.01, 0.02 and 0.1) and the training fraction (0.5, 0.7 and 0.7). For the case of holdout training method the data used for training is the same for all runs, depending on the training fraction used. One run was performed considering these three variables as decision variable ranging in the same interval.

Similarly, for the case of SVMs two different kernels were tested (μ -SVC and C-SVC). Different combinations of training method, learning rate, training fraction and other kernel parameters were tested. The runs performed used the better RPSGA parameters as defined in reference [15]. The comparison between the performances of the different runs was made using the attainment functions methodology [17].

4.2 Influence of Classification Methods Parameters

The results presented in Figures 1, 2 and 3 were obtained using LR with the training method, the learning rate and the training fraction as decision variables. As shown in Figure 1, after 100 generations, the evolution leads to a considerable gain in accuracy while decreasing significantly the number of features needed. On the final population 8 non-dominated solutions exist having, respectively 1, 2, 3, 4, 7, 8, 9 and 10 features, which are identified in Figure 2. Figure 3 shows the values obtained for the training method, the learning rate and the training fraction for the same non-dominated solutions.

Identical results were obtained for the SVM classifier. More results can be seen on www.dep.uminho.pt/agc/results. Finally, the attainment functions methodology was used to compare the performance of the methodology proposed when using different parameters of the classifiers (see the same web site for more details).

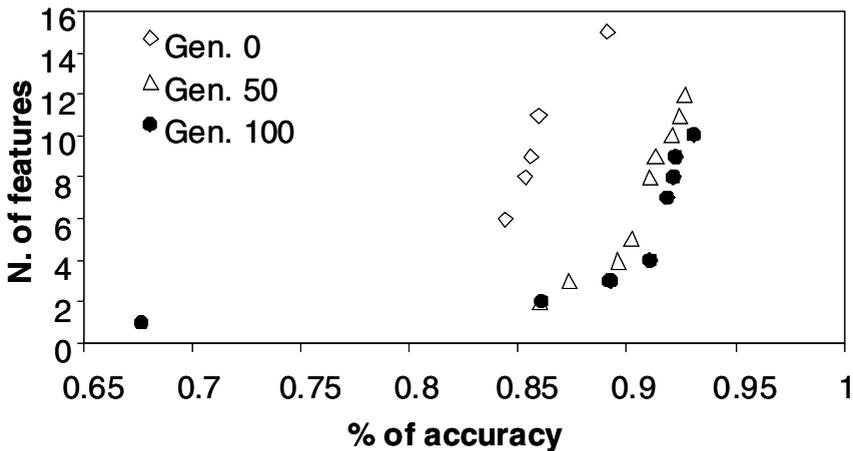


Fig. 1 Pareto fronts for the initial population and after 50 and 100 generations

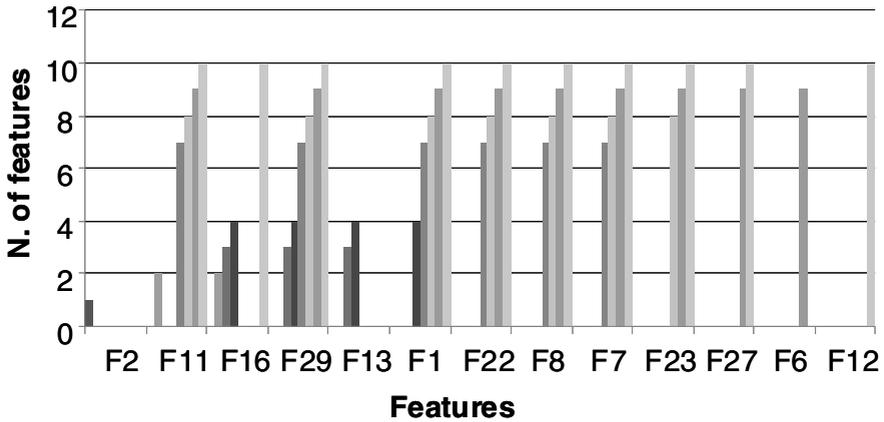


Fig. 2 Features obtained for the non-dominated solutions after 100 generations

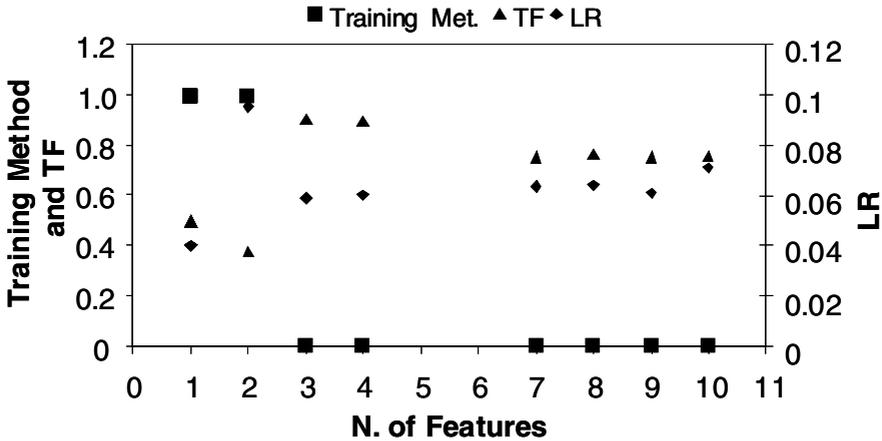


Fig. 3 Training method, learning rate and training fraction values for the 8 non-dominated solutions

5 Conclusions

In the present study the LR and SVM methods with different parameters values have been used. The results obtained allow one to conclude that the proposed methodology is able to solve this problem and, simultaneously, made available information to the decision maker. The information provided concerns not only with the best features to be used but, also, with the best parameters of the classifier.

References

1. Altman, E.: Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *Journal of Finance* 23, 589–609 (1968)
2. Nanda, S., Pendharkar, P.: Linear models for minimizing misclassification costs in bankruptcy prediction. *Intelligent Systems in Accounting, Finance & Management* 10, 155–168 (2001)
3. Charitou, A., Neophytou, E., Charalambous, C.: Predicting Corporate Failure: Empirical Evidence for the UK. *European Accounting Review* 13, 465–497 (2004)
4. Atiya, A.F.: Bankruptcy prediction for credit risk using neural networks A survey and new results. *IEEE Transactions on Neural Networks* 12, 929–935 (2001)
5. Coats, P.K., Fant, L.F.: Recognizing Financial Distress Patterns Using a Neural Network Tool. *Financial Management* 22, 142–155 (1993)
6. Yang, Z.R.: Probabilistic Neural Networks in bankruptcy prediction. *Journal of Business Research* 44, 67–75 (1999)
7. Tan, C., Dihadjo, H.: A study on using artificial neural networks to develop an early warning predictor for credit union financial distress with comparison to the probit model. *Managerial Finance* 27(4), 56–77 (2001)
8. Alfaro-Cid, E., Castillo, P.A., Esparcia, A., Sharman, K., Merelo, J.J., Prieto, A., Mora, A.M., Laredo, J.L.J.: Comparing Multiobjective Evolutionary Ensembles for Minimizing Type I and II Errors for Bankruptcy Prediction. In: 2008 Congress on Evolutionary Computation (CEC 2008), pp. 2907–2913 (2008)
9. Hamdani, T.M., Won, J.-M., Alimi, A.M., Karray, F.: Multi-objective Feature Selection with NSGA II. In: Beliczynski, B., Dzielinski, A., Iwanowski, M., Ribeiro, B. (eds.) ICANNGA 2007. Part I, LNCS, vol. 4431, pp. 240–247. Springer, Heidelberg (2007)
10. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transaction on Evolutionary Computation* 6(2), 181–197 (2002)
11. Agresti, A.: *Categorical Data Analysis*. Wiley-Interscience, New York (2002)
12. Vapnik, V.: *The Nature of Statistical Learning Theory*. Springer, Heidelberg (1995)
13. Gaspar-Cunha, A., Oliveira, P., Covas, J.A.: Use of Genetic Algorithms in Multicriteria Optimization to Solve Industrial Problems. In: Seventh Int. Conf. on Genetic Algorithms, Michigan, USA (1997)
14. Deb, K.: *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley, Chichester (2001)
15. Gaspar-Cunha, A., Covas, J.A.: RPSGAe - A Multiobjective Genetic Algorithm with Elitism: Application to Polymer Extrusion. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) *Metaheuristics for Multiobjective Optimisation*. LNCS, vol. 535, pp. 221–249. Springer, Heidelberg (2004)
16. Gaspar-Cunha, A.: *Modelling and Optimization of Single Screw Extrusion*. PhD Thesis, University of Minho, Guimarães, Portugal (2000), <http://www.lania.mx/~ccoello/EMOO/>
17. da Fonseca, V.G., Fonseca, C., Hall, A.: Inferential performance assessment of stochastic optimisers and the attainment function. In: Zitzler, E., Deb, K., Thiele, L., Coello Coello, C.A., Corne, D. (eds.) *EMO 2001*. LNCS, vol. 1993, pp. 213–225. Springer, Heidelberg (2001)

Dynamic Time Warping of Segmented Time Series

Zoltán Bankó and János Abonyi

Abstract. Providing the most suitable time series representation has always been a crucial factor in time series data mining. The selected approximation does not only determine the tightness of the representation but also the (dis)similarity measure to be used. Piecewise Linear Representation (PLA) is one of the most popular methods when tight representation of the original time series is required; however, there is only one Dynamic Time Warping (DTW) based dissimilarity measure which uses the PLA representations directly. In this paper, a new dissimilarity measure is presented which takes not only the mean of a segment into account but combines it with our recently introduced slope-based approach, which was derived from Principal Component Analysis (PCA).

1 Introduction

A time series is a sequence of values measured as a function of time. These kinds of data are widely used in process engineering, medicine, bioinformatics, chemistry and finance. The increasing popularity of knowledge discovery and data mining tasks for discrete data has indicated the growing need for similarly efficient methods for time series data. These tasks share a common requirement: a (dis)similarity measure or distance has to be defined between the elements of a given database. Moreover, the results of data mining methods — including simple clustering (partitioning the data into coherent but not predefined subsets), classification (placing the data into predefined, labeled groups) and complex decision-making systems — are highly dependent on the used representation of the time series.

Zoltán Bankó · János Abonyi

University of Pannonia, Department of Process Engineering, Veszprem, Hungary

e-mail: abonyij@fmt.uni-pannon.hu

www.fmt.uni-pannon.hu/softcomp

In the past few years, the focus of attention of data mining community has moved from specific tasks such as voice and signature recognition to more general problems such as motif detection and the handling of huge time series databases (i.e. millions of sequences). This does not only mean indexing has become necessary but the motivation behind the application of time series representations and similarity measures has also changed. The representations were used to speed up computationally expensive but very precise DTW or neural network based similarity measures and, in addition, to serve as preprocessing or feature selection steps. Nowadays, this rule seems to be fading and these approximations provide a general and easily indexable scheme with a computationally tractable lower bounding function [1]. However, indexing is often not required due to hardware limitations (embedded systems), precision issues (process control) and small databases or due to developers' motivation [6] but there is always a need for an accurate similarity measure.

In this paper, we present a new DTW-based dissimilarity measure for PLA which does not only consider the mean of each segment but it takes into account the slope of the given segment as well. This property of the introduced algorithm makes it possible to eliminate the major weakness of DTW, i.e. DTW tends to create "singularities". Moreover, the dissimilarity measure can be optimized for the given database by weighting the slope and the mean-based DTW. In other words, the presented measure can be seen as the best compromise between standard PLA-based DTW [8] and recently introduced Derivative time series Segment Approximation (DSA) based DTW [6].

2 Notation

Time series $X = [x(1), x(2), \dots, x(n)]$ and $Y = [y(1), y(2), \dots, x(m)]$ of length n and m are column vectors. The dissimilarity between X and Y is denoted by $d(X, Y)$, where $0 = d(X, Y)$, $d(X, Y) = d(Y, X)$, and $d(X, X) = 0$. The closer the dissimilarity value is to zero, the more similar the processes behind the time series are. Obviously, the dissimilarity is nothing more than a real number, greater than or equal to zero, which expresses the tightness of connection between the processes behind the time series. In practice, it is not required from a dissimilarity measure to satisfy triangular inequality if indexing is not required.

To provide a compact representation of a time series which can be compared with other representation, usually segmentation has to be applied where a number (or a set of numbers) represents each segment according to the selected representation method. For instance, PLA replaces the original data with not equally sized segments of straight lines, i.e. a PLA segment of X is a 4-tuple:

$$\bar{x}(i) = [l_i, r_i, x(l)_i, x(r)_i], \quad (1)$$

¹ A great collection, by Keogh, of popular time series representation and their (sometimes very ingenious) indexing methods can be found at

<http://www.cs.ucr.edu/~eamonn/iSAX/iSAX.html>

where l_i and r_i are the left and the right time coordinates of the i th PLA segment of X , and $x(l)_i$ and $x(r)_i$ are the values of the PLA segment in time coordinates l_i and r_i .

3 Related Work

Besides PLA [8], several representations of time series have been introduced for data mining community such as Piecewise Aggregate Approximation (PAA) [10], Symbolic Aggregate approxImation (SAX) [12], Adaptive Piecewise Constant Approximation (APCA) [3], Episode Segmentation (ES) [5], Discrete Wavelet Transform (DWT) [4] and DSA [6].

PAA divides the original time series into s segments where the length of each segment is the same and they are represented by their means. SAX is based on PAA but the mean values are quantized and SAX symbols are used to denote the quantized levels. APCA also uses the mean of each segment to approximate the original data; however, it can identify segments of different length. PLA uses variable segment length as well but contrary to APCA, each segment is represented by a piecewise linear function instead of a constant value. The recently introduced DSA transforms the original time series to its derivative from and the segmentation is carried out on these derivatives. Finally, the segments are mapped to angular values which represent the average slopes within the segments.

In some cases, the representation determines the (dis)similarity measure to be used (Discrete Fourier/Wavelet Transform - Euclidean distance), whereas other representations allow the application of multiple similarity measures (PLA, PAA, SAX, etc.) to be used.

The common way to compare two time series representations is DTW², which allows the developer to choose the local distance freely, thus it can be formulated to compare the approximations of the segments³. For example, DTW can be applied on the PAA representations as follows: each segment is represented by its mean, hence the DTW algorithm is the same as for the original time series but the computational time is lowered to $O((m/N)^2)$, where m and N are the lengths of the original and the PAA time series, respectively.

Warping the PLA representation is much more difficult. Vullings et al. [14] were the first to use PLA based DTW on ECG data while Keogh and Pazzani [8] gave a generalized distance between two PLA segments $\bar{x}(i)$ and $\bar{y}(j)$, which can be used as a local distance for DTW:

$$d(\bar{x}(i), \bar{y}(j)) = \left(\frac{x(l)_i + x(r)_i}{2} - \frac{y(l)_j + y(r)_j}{2} \right)^2, \quad (2)$$

² Reviewing DTW has been omitted for the sake brevity but the interested reader can find a great review on it in [13].

³ A detailed comparison of most of the above-mentioned representations using DTW can be found in [6].

where $x(l)_i$ and $x(r)_i$ are the values of the i th PLA segment $\bar{x}(i)$ in time coordinates l_i and r_i .

On the one hand, this distance arises from PAA: it compares the means of the corresponding segments, on the other hand, it also build on the tighter PLA representation. However, this tighter representation, more precisely the different length of the segments, introduces a new problem. The normalization factor is usually based on the length of the warping path but PLA, contrary to PAA, does not generate equidistant segments. Thus, Keogh and Pazzani suggested to recursively sum, on the warping path, an additional variable which stores the lengths of the visited segments.

After the (re)introduction of DTW for data the mining community it was clear that *DTW ... can produce pathological results. The crucial observation is that the algorithm may try to explain variability in the Y-axis by warping the X-axis. This can lead to unintuitive alignments where a single point on one time series maps onto a large subsection of another time series. We call examples of this undesirable behavior "singularities"*. Moreover, an additional problem with DTW is that *the algorithm may fail to find obvious, natural alignments in two sequences simply because a feature (i.e. peak, valley, inflection point, plateau, etc.) in one sequence is slightly higher or lower than its corresponding feature in the other sequence* [9]. These problems can be partially eliminated by using special local constraints; however, this solution may prevent the expected warping path to be found. In [9], the authors suggested that instead of raw values (estimated) derivatives should be used, which can provide better results for some databases. Although this method was not extended to segmented representations such as PAA or PLA, F. Gullo et al. introduced a new representation method, called DSA [6], which is also based on the derivatives of the original data but the derivatives are grouped with a modified sliding window algorithm and, finally, each segment is represented by its slope and the timestamp of the last element in the segment plus one.

4 Slope-Based Warping of PLA

Abonyi et al. [2] also used the slope information (i.e. the angular value) to represent a PLA segment. This was motivated by [1] where a new correlation-based segmentation method was presented which was based on the two homogeneity measures of PCA. With the Q reconstruction error — one of the homogeneity measures — a multivariate time series can be segmented as the correlation structure among variables changes. The reconstruction error determines the hyperplanes in each segment, while the goal of the segmentation was to minimize the sum of the squared Euclidean distances between the original and the reconstructed variables in each segment as for PLA. PLA also minimizes the squared Euclidean distance between the original data point and its reconstructed pair (the closest point on the PLA segment). This reconstruction error is shown on a PLA segment in Fig. 1.

Considering that all of the PCA-based similarity measures compare the hyperplanes (principal components) of two multivariate time series, it was logical to

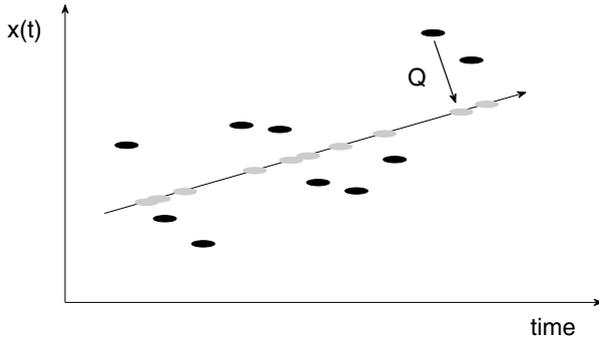


Fig. 1 Time series X (black dots) and its reconstruction using PLA-representation (gray dots)

extend this to PLA where each representation of a PLA-segment (i.e. a straight line) can be handled as the first (and only) principal component of the given segment. This led the authors to represent each PLA-segment with its angle of slope:

$$\bar{x}(i) = \arctan \left(\frac{x(r)_i - x(l)_i}{r_i - l_i} \right) \tag{3}$$

Please note that none of the segments can be perpendicular to the time axis, hence no further restrictions were required. Now, any of the PCA-based similarity measures can be used to compare the segments based on this slope information. The interested reader might notice that the only difference between the PCA-based similarity measures is how they weight the angles between the hyperplanes. In a one-dimensional space, all of them could be reduced to the same equation:

$$d(\bar{x}(i), \bar{y}(j)) = \left| \cos \left(\arctan \left(\frac{\tan(\bar{x}(i)) - \tan(\bar{y}(j))}{1 + \tan(\bar{x}(i)) \tan(\bar{y}(j))} \right) \right) \right|, \tag{4}$$

where $|\cos(\dots)|$ can be replaced by $\cos^2(\dots)$.

Although Eq. 4 would be a perfect similarity measure for hyperplanes of a PCA-projection, where the hyperplanes are the coordinates of the projected space, their signs are not only meaningless but involving them would destroy the precision in case of univariate time series. The PLA-representation of a segment of a univariate time series also represents the original time series in the same space, thus the sign information cannot be neglected. For example, if one compares $\bar{x}(i) = \arctan(57) (\approx 89^\circ)$ and $\bar{y}(j) = \arctan(-57) (\approx -89^\circ \approx 91^\circ)$ as hyperplanes, one will realize that they are almost the same (only 2° is the difference); however, they are completely different from the time series point of view (increasing and decreasing trends). Thus, the local distance of DTW should be a function which can measure the difference between the signed slopes. For [2], the simple squared Euclidean distance was chosen and it has also been used for this paper:

$$d(\bar{x}(i), \bar{y}(i)) = (\bar{x}(i) - \bar{y}(i))^2 \tag{5}$$

The last thing to handle for DTW is the normalization factor (in practice, it is usually the length of the warping path that is chosen). If it can be treated in a way that preprocessing steps are properly executed (i.e. data is filtered and smoothed), the role of the normalization factor is changed. The proper preprocessing ensures that all of the segments are detected due to the underlying process and they are not detected due to the noise; moreover, the minimum number of elements of a segment can also be defined. Therefore, an additional warping is achieved unintentionally if the lengths of the segments are not considered. In other words, the length of the warping path can be used or it can be omitted as it is done in this paper.

4.1 *Combining Slope and Mean-Based Measures*

The results of slope-based warping of PLA, presented in [2], suggested that it is well suited for databases where the standard PLA-based DTW is not precise enough due to singularities or due to the wrong alignment of corresponding but vertically shifted features. However, for the half of the validation databases the standard PLA-based DTW provided better results than the slope-based one. These results motivated us to combine the above-mentioned slope and mean-based similarities. For this paper, a fairly simple but yet straightforward technique has been chosen: both similarity measures have been weighted equally.

5 Validation Results

Based on [7], the presented similarity measure was compared against other methods using the free and widely used datasets and classification algorithm of the UCR Time Series Classification/Clustering Homepage [11]. The datasets were kindly provided by Mr. Keogh on March 7, 2007. Please note, for easier reconstruction of the results, the algorithms were not executed on the four biggest dataset: Face (all), Two Patterns, Wafer and Yoga.

As our combined similarity measure, the competing methods also require one or more parameters to be set. For all segmentation-based methods (grey rows), the time series were segmented to 30 pieces. When DTW is computed, the simplest local constraint is used (i.e. the slope weights equal to one and only three basic moves are allowed), no global constraint (i.e. warping window) was applied and DTW was formalized for PAA and PLA according to [10] and [8], respectively. As a reference, the non-segmented time series were also compared with the Euclidean distance and DTW. Their results are displayed in the rows named ‘Euc.’ and ‘DTW’.

The last four rows contain the results of our new methods: slope-based warping and its combination with its mean-based version as they were applied on PLA segments. They are denoted ‘Angle’ and ‘Mixed’, respectively. For the ‘Mixed.’ method, the classification algorithm of [11] was modified to rank the distances for both similarity measures in an ascending order. These ranks have been summarized, thus the similarity measures have been weighted equally.

Table 1 Results of the UCR time series 1-NN classification algorithm

	50Words	Adiac	Beef	CBF	Coffee	ECG	Face (four)	Fish	Gun-Point	Lightning-2	Lightning-7	OliveOil	Osu Leaf	Swedish Lf.	Synth. Ctrl.	Trace
# of Classes	50	37	5	3	2	2	4	7	2	2	7	4	6	15	6	4
Series Length	270	176	470	128	286	96	350	463	150	637	319	570	427	128	60	275
Train Set Size	450	390	30	30	28	100	24	175	50	60	70	30	200	500	300	100
Test Set Size	455	391	30	900	28	100	88	175	150	61	73	30	242	625	300	100
Euc.	.37	.39	.47	.15	.25	.12	.22	.22	.09	.25	.43	.13	.48	.21	.12	.24
DTW	.31	.40	.50	.00	.18	.23	.17	.17	.09	.13	.27	.13	.41	.21	.01	.00
Euc. on PAA	.36	.41	.50	.07	.25	.13	.17	.21	.09	.25	.34	.13	.47	.21	.01	.29
DTW on PAA	.36	.43	.53	.02	.25	.21	.23	.22	.07	.26	.30	.13	.42	.21	.04	.10
Euc. on PLA	.35	.76	.53	.13	.18	.20	.39	.51	.07	.31	.47	.33	.52	.44	.01	.19
DTW on PLA	.40	.66	.50	.04	.29	.20	.30	.38	.07	.20	.44	.30	.44	.28	.01	.03
Angle Euc.	.46	.71	.67	.64	.32	.23	.72	.46	.05	.39	.78	.40	.70	.51	.67	.17
Angle DTW	.37	.64	.63	.59	.11	.19	.39	.15	.05	.44	.67	.37	.29	.22	.67	.01
Mixed Euc.	.35	.76	.53	.13	.18	.20	.39	.51	.07	.31	.47	.33	.52	.44	.01	.19
Mixed DTW	.34	.57	.60	.17	.18	.15	.28	.15	.05	.28	.38	.33	.29	.17	.07	.00

The results of the suggested 1-NN classification algorithm can be seen in Table 1, in which each cell contains the error rate (proportion of the faulty classified time series). The best results of the segmentation methods were highlighted in **bold** for each database. As it can be seen clearly, the combination of the two methods, denoted by 'Mixed', kept up with the expectations. Its error rate was the lowest seven times and its performance proved to be better as compared to the standard and the slope-based PLA. Moreover, the weights of the similarity measures (i.e. mean and slope-based DTW) were not optimized due to the time limit which suggests that the results can be improved even further.

However, some problematic areas have to be considered. One can realize that the reduction obtained by the segmentation was only about one order of magnitude which ensures the tight representation even with PAA and 30 segments are too much for most databases when PLA is used. Moreover, all of the databases require different number of segments to get the best results.

Thus, we plan to extend our validation in two directions. First, we would like to fine-tune our current experiments by selecting the proper number of segments for each database and compute proper weights for the two similarity measures used in the presented 'Mixed' method. In addition, we plan to compare our method with DSA using these datasets and the validation techniques presented in [6].

6 Conclusion

We presented a new similarity measure for segmented univariate time series by considering the mean and the slope of PLA-representation of a segment. The proposed similarity measure was validated on real-world datasets and we showed that it is worth considering to compare the segments by their slopes in addition to their means. The presented measure offers several advantages. Similarly to DSA, it lowers the chance of faulty mapping of features with DTW and helps to eliminate singularities because it considers slope information. Moreover, there is no need to create a new representation. The existing PLA datasets can be used without further modifications. In addition, it combines the advantages of mean and slope based warping, thus it can be selected how to weight these two different pieces of information.

Acknowledgements. The support from the TAMOP-4.2.2-08/1/2008-0018 (Livable environment and healthier people Bioinnovation and Green Technology research at the University of Pannonia, MK/2) project is gratefully acknowledged.

References

1. Abonyi, J., Feil, B., Németh, S.Z., Arva, P.: Modified gath-geva clustering for fuzzy segmentation of multivariate time-series. *Fuzzy Sets and Systems* 149(1), 39–56 (2005)
2. Bankó, Z., Abonyi, J.: PCA driven similarity for segmented univariate time series. *Hungarian Journal of Industrial Chemistry* (in press)
3. Chakrabarti, K., Keogh, E.J., Mehrotra, S., Pazzani, M.: Locally adaptive dimensionality reduction for indexing large time series databases. *ACM Trans. Database Syst.* 27(2), 188–228 (2002)
4. Pong Chan, K., Fu, A.W.C.: Efficient time series matching by wavelets. In: *ICDE*, pp. 126–133 (1999)
5. Cheung, S.G.: Representation of process trends. *Computers & Chemical Engineering* 14(4-5), 495–510 (1990)
6. Gullo, F., Ponti, G., Tagarelli, A., Greco, S.: A time series representation model for accurate and fast similarity detection. *Pattern Recogn.* 42(11), 2998–3014 (2009)
7. Keogh, E.J., Kasetty, S.: On the need for time series data mining benchmarks: a survey and empirical demonstration. In: *KDD 2002: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 102–111 (2002)
8. Keogh, E.J., Pazzani, M.: Scaling up dynamic time warping to massive datasets. In: Żytkow, J.M., Rauch, J. (eds.) *PKDD 1999. LNCS (LNAI)*, vol. 1704, pp. 1–11. Springer, Heidelberg (1999)
9. Keogh, E.J., Pazzani, M.: Dynamic time warping with higher order features. In: *Proceedings of First SIAM International Conference on Data Mining*, Chicago (2001)
10. Keogh, E.J., Pazzani, M.J.: Scaling up dynamic time warping for datamining applications. In: *Knowledge Discovery and Data Mining*, pp. 285–289 (2000)
11. Keogh, E.J., Xi, X., Wei, L., Ratanamahatana, C.A.: The UCR time series classification/clustering homepage. University of California - Computer Science and Engineering Department, Riverside CA (2006), http://www.cs.ucr.edu/~eamonn/time_series_data/

12. Lin, J., Keogh, E.J., Lonardi, S., Chiu, B.: A symbolic representation of time series, with implications for streaming algorithms. In: DMKD 2003: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, pp. 2–11. ACM, New York (2003)
13. Rabiner, L., Juang, B.H.: Fundamentals of Speech Recognition. Prentice Hall PTR, Englewood Cliffs (1993)
14. Vullings, H.J.L.M., Verhaegen, M.H.G., Verbruggen, H.B.: ECG segmentation using time-warping. In: Liu, X., Cohen, P.R., Berthold, M.R. (eds.) IDA 1997. LNCS, vol. 1280, pp. 275–285. Springer, Heidelberg (1997)

Feature Selection Using Combine of Genetic Algorithm and Ant Colony Optimization

Mehdi Sadeghzadeh, Mohammad Teshnehlab, and Kambiz Badie

Abstract. Feature selection has recently been the subject of intensive research in data mining, especially for datasets with a large number of attributes. Recent work has shown that feature selection can have a positive affect on the performance of machine learning algorithms. The success of many learning algorithms in their attempts to construct models of data, hinges on the reliable identification of a small set of highly predictive attributes. The inclusion of irrelevant, redundant and noisy attributes in the model building process phase can result in poor predictive performance and increased computation. In this paper, a novel feature search procedure that utilizes combining of the Ant Colony Optimization (ACO) and genetic algorithm (GA) is presented. The ACO is a meta-heuristic inspired by the behavior of real ants in their search for the shortest paths to food sources. It looks for optimal solutions by considering both local heuristics and previous knowledge. Genetic algorithm selects the best parameters for ant colony optimization in each step. When this algorithm applied to two different classification problems, the proposed algorithm achieved very promising results.

1 Introduction

Finding the best feature subset for a given problem with N number of features requires evaluating all 2^N possible subsets. The best feature subset also depends

Mehdi Sadeghzadeh

Ph.D. candidate in software engineering, Islamic Azad University,
science and research branch, Tehran, Iran
email: m.sadeghzadeh@mahshahriau.ac.ir

Mohammad Teshnehlab

Associate Professor, K.N.Toosi University of technology, Tehran
email: teshnehlab@eetd.kntu.ac.ir

Kambiz Badie

Associate Professor, Iran Telecom Research Center
email: k_badie@itrc.ac.ir

on the predictive modeling, which will be employed to predict the future unknown values of response variables of interest.

Many factors affect the success of machine learning on a given task. The quality of the data is one such factor, if information is irrelevant or redundant, or the data is noisy and unreliable, the knowledge discovery during training is more difficult.

Feature subset selection is the process of identifying and removing as much of the irrelevant and redundant information as possible. Reducing the dimensionality of the data reduces the size of the hypothesis space and allows algorithms to operate faster and more effectively. [1]

Algorithms for feature selection fall into two broad categories: wrapper that use the learning algorithm itself to evaluate the usefulness of features and filters that evaluate features according to heuristics based on general characteristics of the data. For application to large databases, filters have proven to be more practical than wrappers because they are faster [2].

A feature selection algorithm performs a search through the space of feature subsets and must have following four components [3] :

- A generation procedure to generate the next candidate subset for evaluation, in other words, a search method to explore the possible variable combinations of the search space such as greedy hill climbing that local changes to the current feature subset by adding or deleting a single feature from it [4,5].
- An evaluation function to evaluate the candidate subset.
- A stopping criterion to stop searching through the space of feature subsets. For example, sequential variable selection methods in Multiple Linear Regression models terminate as soon as possible when a variable is found insignificant according to the statistical test.
- A validation procedure to check whether the subset is valid.

A number of search procedure methods have been proposed in the literature. Some of the most famous ones are the stepwise [6], floating search [7], branch-and-bound [8], Genetic Algorithms (GA) [9, 10], and Evolutionary Programming (EP) [11]. we proposed floating-point GA for feature selection (GAFS) [10] that uses classical one point crossover as originally proposed by Holland [12].

2 Ant Colony Optimization for Feature Selection (ACOFS)

Dorigo et. Al [13] proposed an artificial colony of ants algorithm, which was called the Ant Colony Optimization (ACO) meta-heuristic, to solve hard combinatorial optimization problems. The ACO was originally applied to solve the classical travelling salesman problem [20] where it was shown to be an effective tool in finding good solutions.

For a given classification task, the problem of feature selection can be stated as follows: given the original set, F , of n features, find subset S , which consists of m features ($m < n, S \subset F$), such that the classification accuracy is maximized.

The feature selection representation exploited by artificial ants includes :

- n features that constitute the original set, $F = \{f_1, \dots, f_n\}$.
- A number of artificial ants to search through the feature space (na ants).
- τ_i , the intensity of pheromone trail associated with feature f_i , which reflects the previous knowledge about the importance of f_i .
- For each ant j , a list that contains the selected feature subset, $S_j = \{s_1, \dots, s_m\}$.

We propose to use a hybrid evaluation measure that is able to estimate the overall performance of subsets as well as the local importance of features. A classification algorithm is used to estimate the performance of subsets (i.e., wrapper evaluation function). The local importance of a given feature is measured using the correlation based evaluation function, which is a filter evaluation function.

In the first iteration, each ant will randomly choose a feature subset of m features. Only the best k subsets, $k < na$, will be used to update the pheromone trail and influence the feature subsets of the next iteration. In the second and following iterations, each ant will start with $m - p$ features that are randomly chosen from the previously selected k -best subsets, where p is an integer that ranges between 1 and $m - 1$. In this way, the features that constitute the best k subsets will have more chance to be present in the subsets of the next iteration. However, it will still be possible for each ant to consider other features as well. For a given ant j , those features are the ones that achieve the best compromise between pheromone trails and local importance with respect to S_j , where S_j is the subset that consists of the features that have already been selected by ant j . The Updated Selection Measure (USM) is used for this purpose and defined as:

$$USM_i^{S_j} = \begin{cases} \frac{(\tau_i)^\alpha (LI_i^{S_j})^\beta}{\sum_{g \notin S_j} (\tau_g)^\alpha (LI_g^{S_j})^\beta} & \text{if } i \notin S_j \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

Where $LI_i^{S_j}$ is the local importance of feature f_i given the subset S_j . The parameters α and β control the effect of pheromone trail intensity and local feature importance respectively. $LI_i^{S_j}$ is measured using the correlation measure and defined as:

$$LI_i^{S_j} = \frac{|C_{iR}|}{\sum_{f_s \in S_j} |C_{is}|} \quad (2)$$

Where $|C_{iR}|$ is the absolute value of the correlation between feature $i (f_i)$ and the response (class) variable R , and $|C_{is}|$ is the absolute value of the inter-correlation between feature $i (f_i)$ and feature $s (f_s)$ that belongs to S_j .

Below are the steps of the algorithm:

1. Initialization:

- Set $\tau_i = cc$ and $\Delta T_i = 0, (i = 1, \dots, n)$, where cc is a constant and $\Delta \tau_i$ is the amount of change of pheromone trail quantity for feature f_i .
- Define the maximum number of iterations.
- Define k , where the k -best subsets will influence the subsets of the next iteration.
- Define p , where $m - p$ is the number of features each ant will start with in the second and following iterations.

2. If in the first iteration,

- For $j=1$ to na ,
 - Randomly assign a subset of m features to S_j .
- Go to step 4.

3. Select the remaining p features for each ant:

- For $mm = m - p + 1$ to m ,
 - For $j=1$ to na ,
 - Given subset S_j , Choose feature f_i that maximizes $USM_i^{S_j}$.
 - $S_j = S_j \cup \{f_i\}$.

- Replace the duplicated subsets, if any, with randomly chosen subsets.

4. Evaluate the selected subset of each ant using a chosen classification algorithm:

- For $j=1$ to na ,
 - Estimate the Error (E_j) of the classification results obtained by classifying the features of S_j .

- Sort the subsets according to their E . Update the minimum E (if achieved by any ant in this iteration), and store the corresponding subset of features.
5. Using the feature subsets of the best k ants, update the pheromone trail intensity:
- For $j=1$ to k , /* update the pheromone trails */

$$\Delta\tau_i = \begin{cases} \frac{\max_{g=1:k}(E_g) - E_j}{\max_{h=1:k}(\max_{g=1:k}(E_g) - E_h)} & \text{if } f_i \in S_j \\ 0 & \text{Otherwise} \end{cases} \quad (3)$$

$$\tau_i = \rho \cdot \tau_i + \Delta\tau_i \quad (4)$$

Where ρ is a constant such that $(1 - \rho)$ represents the evaporation of pheromone trails.

6. If the number of iterations is less than the maximum number of iterations, or the desired E has not been achieved, initialize the subsets for next iteration and goto step 3 :
- For $j=1$ to na ,
 - From the features of the best k ants, randomly produce $m - p$ feature subset for ant j , to be used in the next iteration, and store it in S_j .
 - Go to step 3.

3 Combining Ant Colony Optimization and Genetic Algorithm for Feature Selection (GAACOFS)

In combining ant colony algorithm and genetic algorithm, we use from population with 12 individuals that contains values for α, β, ρ . Since these values are real and between 0 until 1, we use from continuous genetic algorithm. Before the pheromone updating, the best individual is selected with genetic algorithm, and the best parameters are passed to ant colony algorithm. Mutation rate is 0.2 and selection coefficient is 0.5, i.e. half best individuals in population are kept for next generation. The initial value of α, β, ρ are 0.6, 0.7 and 0.75.

4 Experiments

In order to evaluate the effectiveness of these algorithms as a feature selector for common machine learning algorithms, experiments were performed using seven

data sets from the UCI collection [14] and three additional artificial domains were borrowed from work by Langly and Sage [15] that each have 3 relevant features to which a further 27 irrelevant features have been added.

The criterion of evaluating features used for FFS, BFS, PFFS, and BBFS methods is 1-nearest neighbour leave-one-out classification performance, and 60 percent of data were used to train and 40 percent used to test them.

The parameters of ACOFS algorithm are assigned the following values:

- $\alpha = \beta = 1$, which basically makes the trail intensity and local measure equally important.
- The number of ants, $na = 200$, and the maximum number of iterations is 20.
- $k = 50$. Thus, only the best $na / 4$ ants are used to update the pheromone trails and effect the feature subsets of the next iteration.
- If $m < 9$ then $p = \lfloor m / 2 \rfloor$, otherwise $m - p = \max(m - 4, \text{round}(0.6 \times m))$, where p is the number of the remaining features that need to be selected in each iteration. It can be seen that p will be equal to 4 if $m \geq 9$. The logic behind this is that evaluating the importance of features locally becomes less reliable as the number of selected features increases. In addition, this will reduce the computational cost especially for large values of m .
- The initial value of trail intensity $cc = 1$, and the trail evaporation is 0.25, i.e., $\rho = 0.75$.
- The Error of a k-nearest neighbour classifier trained with chosen randomly 60 percent of data is used to evaluate the performance of the selected subsets in each iteration.

Two machine learning algorithms representing two diverse approaches to learning were used in the experiments, a probabilistic learner (Naive Bayes), and an instance-based learner (KNN). The percentage of correct classification were calculated for each algorithm- data set combination before and after feature selection, and showed in Tables 1 and 2. We use from 10 fold cross-validation on the train set for calculating average correct classification rates. Bold values show where one method is better than the others method. From these tables, we see that in some cases intelligent methods are better than other feature selection methods. The number of selected features in GAFS, ACOFS, and GAACOFS is equal to number of features is used in FFS.

Feature selection improves the performance of Naive bayes and k-nn classifiers In some of natural domains and artificial domains.

In artificial domains intelligent methods have successfully removed the 27 irrelevant attributes from the first two boolean domains (B1 & B2), but As expected they are not effective on the third of the boolean domains (B3). Due to the high degree of feature interaction in this domain, none of the relevant features in isolation can be distinguished from the irrelevant ones.

Table 1 The percentage of correct classification (Naïve bayes)

Data set	Without	FFS	BFS	PFFS	BBFS	GAFS	ACOFS	GAACOFS
Contact-lenses	70.83	87.5	87.5	87.5	87.5	87.5	87.5	87.5
Diabetes	76.3	74.09	73.05	74.09	76.43	75.65	76.43	76.43
Glass	49.07	54.67	45.33	54.67	36.45	48.13	43.93	48.59
Heart-statlog	83.7	77.78	81.85	81.85	70	85.56	80.37	83.70
Iris	96	96	96	96	96	96	96	96
Vehicle	44.80	47.75	52.13	49.41	38.06	44.32	50.47	41.96
Zoo	95.05	92.08	91.09	92.08	73.27	93.07	92.08	92.08
B1	94.17	100	100	100	87.5	100	100	100
B2	90.83	100	100	100	75	100	100	100
B3	74.17	75	75	75	75	75	75	75

Table 2 The percentage of correct classification (KNN)

Data set	Without	FFS	BFS	PFFS	BBFS	GAFS	ACOFS	GAACOFS
Contact-lenses	79.17	87.5	87.5	87.5	87.5	87.5	87.5	87.5
Diabetes	70.18	71.48	68.49	71.48	68.75	70.18	66.80	70.18
Glass	70.56	64.02	67.29	64.02	65.89	69.92	70.18	70.18
Heart-statlog	75.19	72.22	81.11	81.11	61.48	78.15	76.30	76.67
Iris	95.33	96	96	96	96	96	96	96
Vehicle	69.86	70.09	70.80	73.40	54.96	60.05	71.16	63.00
Zoo	96.04	97.03	96.04	97.03	73.27	90.01	94.06	95.05
B1	87.5	100	100	100	87.5	100	100	100
B2	69.17	100	100	100	75	100	100	100
B3	73.33	100	100	100	75	70.83	74.17	75

5 Conclusion

In this paper, we presented a novel feature selection search procedure based on the combining Ant Colony Optimization meta-heuristic and genetic algorithm (GAACOFS) and then compared performance of it with an ant colony algorithm that use heuristic function for evaluating selected features and genetic based feature selection algorithm that operates independently of any induction algorithm. The feature selection search procedure based on the Ant Colony Optimization utilizes both local importance of features and overall performance of subsets to search through the feature space for optimal solutions. When used to select features in presented datasets, the proposed algorithm in most cases outperformed

other feature selection methods. Results show GAACOFS algorithm selects a good subset of features that are useful to common machine learning algorithms by improving their accuracy and making their results easier to understand especially in data sets with irrelevant or redundant features. It is obvious that in feature selection, improvement in correct classification rate depends on the correlation between features, and hence depends on data set. Therefore in data sets with uncorrelated features and without irrelevant features, feature selection may result to decreasing of correct classification rate. Proposed evaluation function cannot identify strongly interacting features such as in a parity problem. One approach might be to model higher order dependencies, that is, correlation between pairs of features and the class.

References

- [1] Hall, M.A.: Correlation-based Feature selection for Machine Learning, Ph.D. Thesis, Department of Computer Science. Hamilton. The University of Waikato, New Zeland (1999)
- [2] Kohavi, R., John, G.H.: The Wrapper Approach. In: Liu, H., Motoda, H. (eds.) Feature Selection for Knowledge Discovery and Data Mining, pp. 33–50. Kluwer Academic Publishers, Dordrecht (1998)
- [3] Dash, M., Liu, H.: Consistency-based search in feature selection. *Artificial Intelligence* 151, 155–176 (2003)
- [4] Deng, K., Moore, A.: On the Greediness of Feature Selection Algorithms. In: Van den Herik, H.J., Weijters, T. (eds.) International Conference of Machine Learning (ICML 1998), University Maastricht, The Netherlands (1998)
- [5] Skalak, D.: Prototype and Feature Selection by Sampling and Random Mutation Hill Climbing Algorithms. In: Eleventh International Machine Learning Conference, pp. 293–301. Morgan Kaufmann, New Brunswick (1994)
- [6] Klitter, J.: Feature set search algorithms. In: Chen, C.H. (ed.) Pattern Recognition and signal Processing, Sijhoff and Noordhoff, the Netherlands (1978)
- [7] Paudil, P., Novovicova, J., Klittler, J.: Floating search methods in feature selection. *Pattern Recognition Letters* 15, 1119–1125 (1994)
- [8] Narendra, P.M., Fukunaga, K.: A branch and bound algorithm for feature subset selection. *IEEE Transaction on Computers* C-26, 917–922 (1977)
- [9] Yang, J., Honavar, V.: Feature subset selection using a genetic algorithm. *IEEE Transactions on Intelligent Systems* 13, 44–49 (1998)
- [10] Sadeghzadeh, M., Shenasa, M.H.: Correlation Based Feature Selection Using Genetic Algorithms. In: Proceedings of the Third International Conference on Modeling, Simulation and Applied Optimization, Sharjah, U.A.E. (2009)
- [11] Sadeghzadeh, M., Shenasa, M.H.: Correlation Based Feature Selection Using Evolutionary Programming. In: Proceedings of the Twelfth IASTED International Conference on Artificial Intelligence and Soft Computing, Palma de Mallorca, Spain, pp. 185–190 (2008)

- [12] Holland, J.H.: Adaption in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
- [13] Dorigo, M., Maniezzo, V., Colorni, A.: Ant Systems: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B* 26, 29–41 (1996)
- [14] Blake, C., Keogh, E., Merz, C.J.: UCI Repository of Machine Learning Data Bases, University of California, Department of Information and Computer Science, Irvine, CA (1998), <http://www.ics.uci.edu/~mlean/MLRepository.html> (Accessed June 20, 2008)
- [15] Langky, P., Sage, S.: Scaling to domains with irrelevant features. In: Greiner, R. (ed.) *Computational Learning Theory and Natural Learning Systems*. MIT Press, Cambridge (1994)

Part IV

**Soft Computing for Image and Signal
Processing, and Pattern Recognition**

Fully Automatic Boundary Extraction of Coronary Plaque in IVUS Image by Anisotropic Diffusion and T-S Type Fuzzy Inference

Takanori Koga, Shohei Ichiyama, Eiji Uchino, Noriaki Suetake,
Takafumi Hiro, and Masunori Matsuzaki

Abstract. This paper describes a fully automatic plaque boundary extraction method for the intravascular ultrasound image by using anisotropic-diffusion-based pre-processing and Takagi-Sugeno (T-S) type fuzzy inference. In the pre-processing, areas for plaque boundary extraction are automatically searched and found by image processing and some heuristic rules. In the found areas, the objective boundaries are then extracted by T-S type fuzzy inference. The present method has reduced substantially the workload of medical doctors.

1 Introduction

Intravascular ultrasound (IVUS) method [1] is one of the tomographic imaging techniques. The IVUS method provides a real-time cross-sectional image of a coronary artery in vivo, and is employed for a visualization of an atherosclerotic plaque for a diagnosis of the acute coronary syndromes (ACS). In the diagnosis of ACS, a precise boundary extraction of plaque is strongly required.

Takanori Koga

Fuzzy Logic Systems Institute, 680-41 Kawazu, Iizuka,

Fukuoka 820-0067, Japan

e-mail: koga@flsi.or.jp

Shohei Ichiyama · Eiji Uchino · Noriaki Suetake

Graduate School of Science and Engineering, Yamaguchi University,

1677-1 Yoshida, Yamaguchi 753-8512, Japan

e-mail: ichiyama@ic.sci.yamaguchi-u.ac.jp,

uchino@yamaguchi-u.ac.jp, suetake@sci.yamaguchi-u.ac.jp

Takafumi Hiro

School of Medicine, Nihon University, 30-1 Oyaguchi Kami-cho,

Itabashi-ku 173-8610, Japan

Masunori Matsuzaki

Graduate School of Medicine, Yamaguchi University, 2-14-1 Tokiwadai,

Ube 753-8505, Japan

Medical doctors usually trace the boundary of plaque by hand and then evaluate its area for many IVUS images. The extraction of plaque boundary is a hard and time-consuming work for medical doctors. Furthermore, the boundary extraction work is very troublesome because IVUS image is so grainy due to the blood speckle noise. Hence, an automatic precise boundary extraction of plaque is strongly desired.

In the representative boundary extraction methods to date, the seed points are given by a medical doctor, and those points are interpolated by a spline function [2]. Those methods can reduce slightly the workload of a medical doctor. However, the accuracy of interpolation, i.e., the detection accuracy of plaque boundary, is considerably affected by the number of the seed points and/or where those points are placed.

On the other hand, the methods based on automatic search algorithms, e.g., an active contour (snake) model, genetic algorithm, and so on, have been proposed so far [3]. However, those methods involve some iterative processes and take plenty of computing time. To meet with a practical demand in clinic, a quick and fully automatic boundary extraction of plaque is definitely necessary.

The authors have proposed a plaque boundary extraction method [4] by employing a statistical discriminant measure (being called separability [5]) and Takagi-Sugeno (T-S) fuzzy model [6]. The method realizes a fast, precise, robust, and semiautomatic extraction of the plaque boundary. The method does not include any time-consuming iterative processes as in the conventional methods. This method [4] has however the following drawback; not less than a few seed points have to be given by a medical doctor in order to decide a plaque boundary extraction area. It is essentially a semiautomatic method.

In this study, we propose a fully automatic boundary extraction method. In the present method, the plaque boundary extraction area is automatically searched and found by an edge-preserved smoothing using anisotropic diffusion, weighted separability, and some heuristic rules. Then in the found area, the objective boundaries are extracted by T-S type fuzzy inference.

2 Intravascular Ultrasound (IVUS) Image and Plaque Boundary Extraction

The image shown in Fig. 1(a) is called a “B-mode image.” This is a cross-sectional image of coronary artery obtained by IVUS method. The following two boundaries are extracted. One is a luminal boundary (LB) between the lumen and the plaque, and the other is an adventitial boundary (AB) between the plaque and the vascular wall as shown in Fig. 1(a).

In this paper, the fuzzy-inference-based boundary extraction method proposed in [4] is improved. In that method, a plaque boundary is approximated by piecewise polynomials inferred by Takagi-Sugeno (T-S) fuzzy model based on the given seed points. The boundary extraction procedure is briefly summarized as follows:

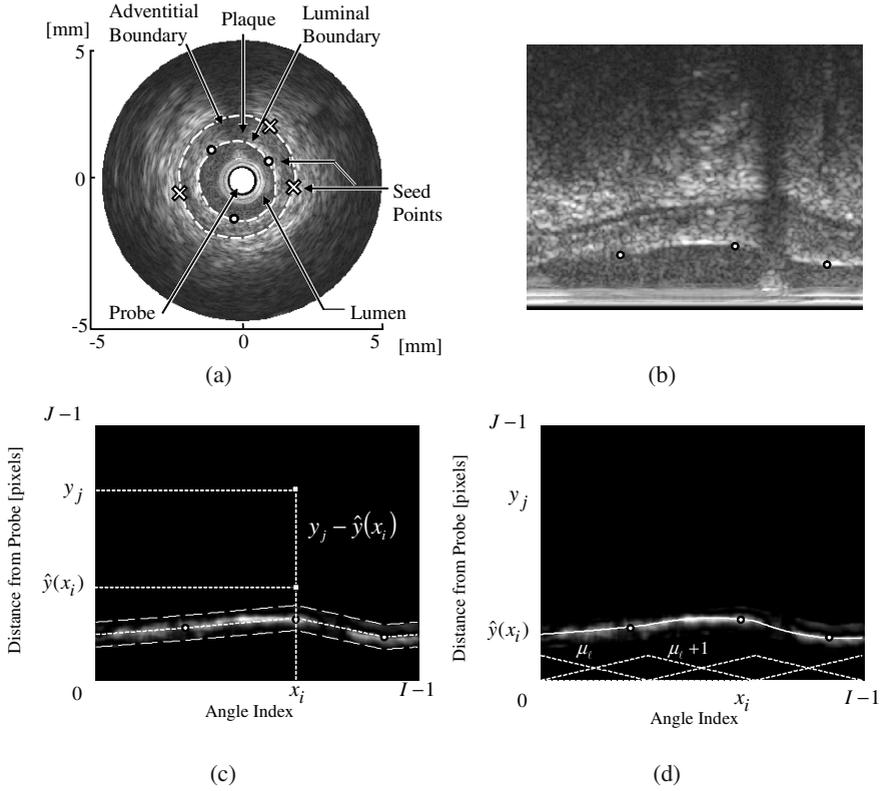


Fig. 1 IVUS B-mode image and plaque boundary extraction scheme. (a) The dotted lines show luminal boundary (LB) and adventitial boundary (AB) to be extracted. (b) Transformed B-mode image into the Cartesian coordinates, and seed points for extraction of LB. (c) A primitive boundary and a specified search area, within the two thin dotted lines, for the objective boundary to be extracted. (d) The extracted luminal boundary (LB) by T-S type fuzzy inference.

1. Seed points are placed on the B-mode image by a medical doctor. This image is then transformed into the Cartesian coordinates as shown in Fig. 1(b).
2. A statistical discriminant measure of separability of image [5] is calculated. The separability η_s of pixel $s = (i, j)$ takes a large value around the regional edge of the image. The brightness of each pixel in the discriminant image is a value of separability η_s for that pixel. Fig. 1(c) shows a discriminant image of Fig. 1(b). That is, a chain of bright pixels can then be a candidate of a boundary.
3. The seed points are first linearly interpolated on the Cartesian coordinates to obtain a primitive boundary (the bold dotted line in Fig. 1(c)). The true boundary is then searched by starting from this primitive boundary. The boundary extraction area is within the two thin dotted lines in Fig. 1(c).

4. The objective true boundary is inferred by T-S fuzzy model. The boundary is piecewise approximated by the polynomials in the Cartesian coordinates by a series of the following fuzzy if-then rules:

$$\text{IF } x_i \text{ is } A_\ell \text{ THEN } f_\ell(x_i) = a_\ell x_i + b_\ell,$$

where A_ℓ is a fuzzy set with membership function μ_ℓ as shown in Fig. 1(d). x_i corresponds to an angle index, and $f_\ell(x_i)$ is a linear function. The ℓ -th rule thus stands for a piecewise approximation of a plaque boundary by a linear function in the interval $[t_{\ell-1}, t_{\ell+1}]$. The inferred boundary $\hat{y}(x_i)$ is given by:

$$\hat{y}(x_i) = \mu_\ell(x_i)f_\ell(x_i) + \mu_{\ell+1}(x_i)f_{\ell+1}(x_i). \quad (1)$$

The optimum coefficients a_ℓ^* and b_ℓ^* of the ℓ -th fuzzy if-then rule are determined with use of the weighted least square method (WLSM) so as to minimize the following weighted error criterion:

$$E = \sum_{j=0}^{J-1} \sum_{i=0}^{I-1} \eta_{ij}^2 \{y_j - \hat{y}(x_i)\}^2, \quad (2)$$

where η_{ij} is a separability of pixel $\mathbf{s} = (i, j)$ [5].

3 Proposed Method

In our previous method [4], the area for boundary extraction is restricted around the linear interpolation curve connecting the seed points (bold dotted line in Fig. 1(c)). In the present method, we propose an automatic search method for the area of boundary extraction without using any seed points. The following are the concrete processes of each step shown in Figs. 2(a)-(h).

At first, edge-preserved smoothing using anisotropic diffusion is performed on the target image of Fig. 2(a). In this study, the anisotropic diffusion filter proposed by Perona and Malik [7] is employed to realize an edge-preserved smoothing of image. The basic idea behind the Perona-Malik Diffusion (PMD) is to get an increasingly smoothed image as shown in Fig. 2(b) from the target image of Fig. 2(a).

In order to distinguish the edge of plaque from that of a speckle pattern in an IVUS image, the proposed method employs a statistical discriminant measure of weighted image separability. This is a modification of [5]. It detects the candidates of the inner (lower) and the outer (upper) boundaries of plaque by considering the following two conditions peculiar to an IVUS image. Those are: 1) intensity in the outside area of a luminal boundary (LB) tends to be bigger than that in the inside area of LB; 2) intensity in the outside area of an adventitial boundary (AB) tends to be bigger than that in the inside area of AB. The weighted separability for a pixel $\mathbf{s} = (i, j)$ is defined by:

$$\eta_{\mathbf{s}}^w = \eta_{\mathbf{s}} \times \left(\frac{I_{\max} - \bar{I}_1}{I_{\max}} \times \frac{\bar{I}_2}{I_{\max}} \right)^2, \quad (3)$$

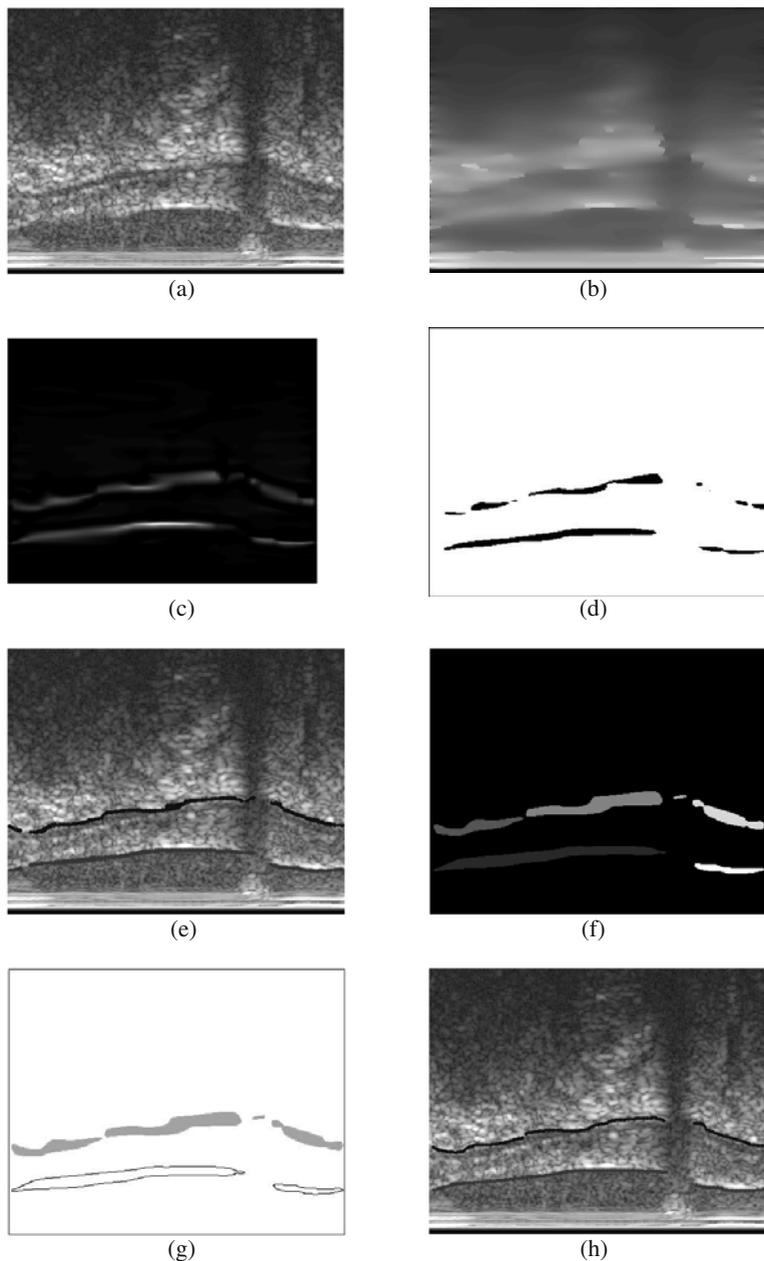


Fig. 2 Automatic detection of seed points. (a) Target image. (b) Edge-preserved smoothing image by PMD. (c) Discriminant image of (b) by weighted separability. (d) Binary image obtained by threshold processing for (c). (e) Candidates of the seed points. (f) Regions for getting the final seed points. (g) Classified regions of (f) for LB or AB. (h) Final seed points automatically detected.

where I_{max} is a maximum intensity on the whole IVUS image. \bar{I}_1 and \bar{I}_2 are the means of the intensities in the two discriminative regions 1 and 2, respectively [4]. Fig. 2(c) shows a discriminant image of Fig. 2(b) by using η_s^w . Each pixel of the image represents the weighted separability η_s^w . When η_s^w is large, high brightness is assigned to its corresponding pixel. The weighted separability takes a large value around the regional edge of an IVUS image. That is, a chain of bright pixels can then be a candidate of a plaque boundary.

Fig. 2(d) shows a binary image obtained by the threshold processing for the image of Fig. 2(c). Let the threshold value of the weighted separability be ζ . By changing the value of ζ from ζ_{low} to ζ_{high} at regular intervals, a binary image with each threshold value of ζ such as Fig. 2(d) is obtained.

The points on the central line of each black-colored region of Fig. 2(d) are regarded as candidates of the seed points. The candidates are detected by line scanning in parallel to the x-axis. When two regions are detected by the line scanning, the centers of regions are regarded as the candidates. The candidates of the seed points for all the binary images are superimposed in Fig. 2(e).

An optimum threshold value of ζ has to be set to obtain the best binary image of Fig. 2(d) for detecting the final seed points. In this process, the binary images are evaluated by using the following score:

$$E = Grade1 \times Grade2 \times Grade3, \quad (4)$$

where *Grade1*, *Grade2* and *Grade3* are calculated by the functions shown in Figs. 3(a), (b) and (c), respectively. Those functions are empirically determined by analyzing hundreds of real IVUS images. An image, say Fig. 2(f), with the maximum score of E is selected.

The color-coded regions shown in Fig. 2(f) are then classified either to the region for LB or that for AB as shown in Fig. 2(g) by majority decision by using the class information of the candidates of the seed points.

Finally, the points on the central line of each region of Fig. 2(g) are regarded as the final seed points to get Fig. 2(h). The selected seed points are linearly interpolated, and the boundary extraction areas are allocated around this linearly

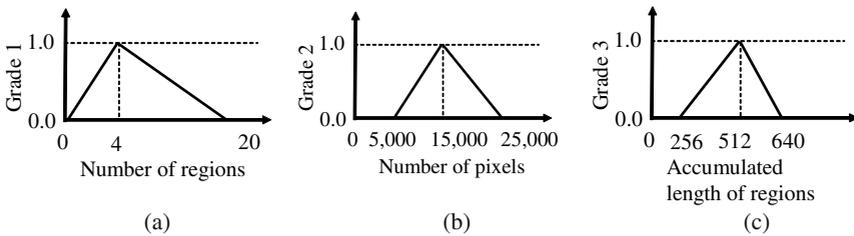


Fig. 3 Three functions to select the best binary image for detecting the final seed points. (a) Function to evaluate the number of regions. (b) Function to evaluate the number of pixels in the regions. (c) Function to evaluate the accumulated length of the regions.

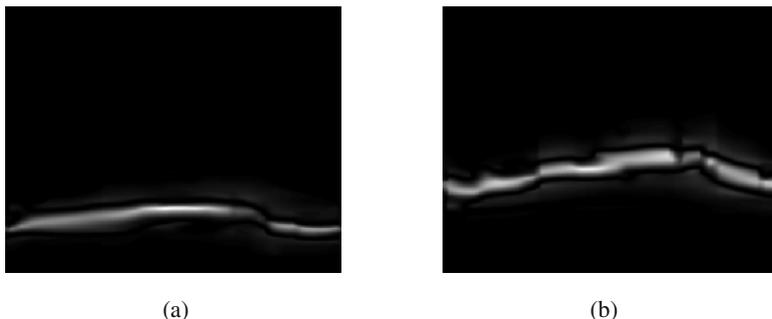


Fig. 4 The plaque boundary extraction areas allocated by the present method. (a) The allocated area for luminal boundary (LB). (b) The allocated area for adventitial boundary (AB).

interpolated curve with weights of Gaussian kernel function. The allocated areas are shown in Fig. 4.

By using Figs. 4(a) and (b), the plaque boundary is extracted by T-S type fuzzy inference.

4 Experimental Results and Discussion

The proposed method is applied to real IVUS B-mode images. Extraction results are compared to those by spline function and by our previous method [4]. In those experiments, the threshold values of ζ_{low} and ζ_{high} are set to be 0.1 and 0.2, respectively. The binary images are generated by changing ζ at 0.02 intervals. The other settings are the same as those in our previous paper [4].

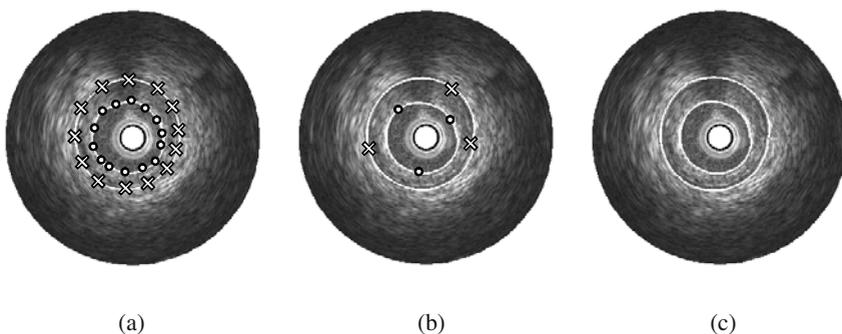


Fig. 5 Plaque boundary extraction results. \times : Given seed points for adventitial boundary (AB). \circ : Given seed points for luminal boundary (LB). (a) Extracted boundaries by spline function. (b) Extracted boundaries by our previous method. (c) Extracted boundaries by the present method.

Figs. 5(a)-(c) show the plaque boundary extraction results for the IVUS image. The results show that the proposed method works very well by no means inferior to the conventional methods, even if without using any seed points. Particularly, the spline function method requires many seed points though the accuracy of boundary extraction is good.

The root mean square errors (RMSEs) between the true boundary $y^*(x_i)$ and the extracted one $\hat{y}(x_i)$, and the standard deviations (SDs) calculated from 60 IVUS images are shown in Table 1. The true boundary is calculated by a parametric spline-interpolation using a lot of seed points. It is observed that the proposed method, fully automatic and requires no seed points given in advance, gives almost equal extraction results to our previous method which needs a few seed points given by a medical doctor.

With those results, the effectiveness of the proposed method has been verified.

Table 1 Evaluation for the plaque boundary extraction results.

	(μm)					
	Spline function		Previous method		Present method	
	LB	AB	LB	AB	LB	AB
RMSE	14.5	16.9	36.5	52.1	37.5	63.2
SD	14.3	16.5	34.5	31.1	34.9	33.3

5 Conclusions

We have proposed a fully automatic plaque boundary extraction for an IVUS image. In the present method, the plaque boundary extraction area is automatically searched and found by using anisotropic diffusion, weighted separability, and some heuristic rules. T-S type fuzzy inference is used to extract the plaque boundary within that area. The present method has shown an almost equal extraction performance to our previous method without bothering medical doctors.

Acknowledgements. This work was supported in part by “Knowledge Cluster Initiative,” for 2004-2009, funded by Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

1. Nicholls, S.J., et al.: Intravascular ultrasound in cardiovascular medicine. *Circulation* 114, 54–59 (2006)
2. Ferguson, J.: Multivariable curve interpolation. *Journal of the Association for Computing Machinery* 11, 221–228 (1967)
3. Klingensmith, J.D., et al.: Assessment of coronary compensatory enlargement by three-dimensional intravascular ultrasound. *International Journal of Cardiac Imaging* 16, 87–98 (2000)

4. Uchino, E., et al.: Fuzzy inference-based plaque boundary extraction using image separability for intravascular ultrasound image of coronary artery. *Electronics Letters* 45, 451–453 (2009)
5. Fukui, K.: Edge extraction method based on separability of image features. *IEICE Transactions on Information and Systems* E78–D, 1533–1538 (1995)
6. Takagi, T., et al.: Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics* 15, 116–132 (1985)
7. Perona, P., et al.: Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12, 629–639 (1990)

Cost Effective 3D Localization for Low-Power WSN Based on Modified Sammon's Stress Function with Incomplete Distance Information

Lakshmesha Rao, Kuncup Iswandy, and Andreas König

Abstract. A crucial aspect of mobile 3D wireless sensor network application is cost effective localization of individual nodes. GPS based localization is costly and not feasible in many applications. Among GPS free localization methods, Multi-Dimensional Scaling (MDS) based localization methods has been well accepted solution for localization problem, due to low localization error. MDS based localization uses the connectivity information among the nodes to find the positions of nodes in the network. MDS needs a full distance matrix as input. Various methods, e.g., Dijkstra's algorithm, have been used to generate full distance matrix from incomplete distance matrix obtained from the connectivity information of nodes at the price of $O(n^2)$ added computational complexity. This paper presents a novel idea of trying to generate the position information at lowered computational cost and correspondingly reduced power requirement without need for full distance matrix. We will present a modified cost function and demonstrate Sammon's Mapping with standard gradient descent techniques as well as Genetic algorithms and Particle Swarm Optimization. Our experimental results show, that the proposed localization approach can lead to savings in numerous practical cases. With regard to mapping error reduction, standard GA and PSO so far did not offer a improvement. In future work variations of GA and PSO will be investigated.

1 Introduction

A wireless sensor network (WSN) is a network consisting of spatially distributed devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations. These sensor nodes have many industrial and consumer applications,

Lakshmesha Rao · Kuncup Iswandy · Andreas König
Institute of Integrated Sensor Systems, TU Kaiserslautern, 67663 Kaiserslautern
e-mail: rao@rhrk.uni-kl.de, {kuncup, koenig}@eit.uni-kl.de

e.g., in agriculture and ambient intelligence. In the standard case WSN is employed in planar arrangement. More recently, 3D deployment scenarios, e.g., underwater network (USN), are emerging. Sensor network commonly only posses small batteries, e.g., see mica motes, thus any overhead and power consumption due to localization should be avoided.

Sensing data becomes particularly useful if the context of the location is available. Thus an important aspect in most of the WSN application is the localization of the individual nodes [1]. Localization can be defined as method of estimating position or spatial coordinates of WSN [11]. Localization without using Global Positioning System (GPS) has always been a challenge in the wireless sensor networks, as GPS is a costly and impractical solution in many applications. Thus GPS free localization has received much research attention from scientific community.

Among the various methods of localization developed, range based localization algorithms give the most accurate information on sensor positions. Within available range based methods, MDS-MAP has been shown to have lowest localization error [6]. In our research MDS-MAP is tried for the case of 3-dimensional sensor deployment. One of the main requirement of the MDS based algorithms presented till now has been the need for Full Distance Matrix (FDM) as the input. In reality this FDM is not available as nodes in wireless network have to use multi-hop based communication and hence are not fully connected. This leads to Incomplete Distance Matrix (IDM) where the nodes without connections are represented by a common flag. IDM also provides the connectivity information. FDM is derived from the connectivity information (IDM) available between the nodes in the network. One of the important method for generating the FDM is Dijkstra's shortest path algorithm [3].

There are three sources of errors while solving localization problem using MDS-MAP. First source is the errors introduced by ranging measurements (RSSI, Ultrasound etc.). Second source is the error introduced while FDM is generated using the shortest path algorithms, e.g., Dijkstra's one. Third source is the mapping error of the employed algorithm itself. Our idea is to increase the localization accuracy by trying to remove the error introduced by shortest path algorithms. Moreover the computational complexity of shortest path algorithm is high, e.g., computational complexity of Dijkstra's algorithm is $O(n^2)$, for n number of nodes, which has obvious implications for power consumption. Thus IDM based localization methods would be computationally cost effective. Computational cost effectiveness is more pronounced when the wireless sensor nodes are mobile. Striving for the overall error reduction we have investigated NLM with IDM based on a modified cost function. Further, to potentially reduce the mapping error itself, in addition to gradient descent, genetic algorithms and particle swarm optimization have been employed in the case study.

Next section will explain the MDS-MAP based approach. Section 3 explains the proposed nonlinear mapping using newly introduced cost function, as well as issues of PSO and GA for MDS. Simulations using different experimental scenarios and results obtained are described in Section 4, before concluding in Section 5.

2 MDS-MAP

Use of multi-dimensional scaling for the purpose of localization was proposed by Shang et al. [10]. MDS-MAP is a centralized localization algorithm, which uses the connectivity information to achieve localization. This algorithm is well proven for 2D localization and was depicted to be the best algorithm for 2-dimensional deployment scenario [6]. MDS-MAP works based on the idea of distance preserved between the nodes. From the distance map the original positions are found out, given that we have enough anchor nodes.

The process of localization has three steps:

1. **ShortestPath:** Compute shortest paths between all pairs of nodes in the region of consideration from IDM. Dijkstra's algorithm has been used to construct FDM using shortest path distances.
2. **MDS:** Apply Classical metric based MDS (CMDS) to the FDM obtained in previous step. Retain the 3 eigenvectors having largest eigenvalues to construct a 3-D relative map.
3. **MapBack:** Given sufficient anchor nodes, at least 4, transform the relative map to an absolute map based on the absolute positions of anchors.

It is an open issue, whether the potentially error prone FDM calculation is really necessary in general, or whether useful mappings can be obtained straight from IDM at lower computational cost and power consumption. In next section methods for generating the co-ordinates of nodes without need for full distance matrix will be discussed.

3 Sammon's Non-linear Mapping

Classical MDS scheme is a linear mapping of data trying to preserve the interpoint distance between data while translating from higher dimension, \mathbb{R}^p , to lower dimension, \mathbb{R}^d . Non-Linear method of Mapping (NLM) data was introduced by Sammon in 1969 [4]. Mapping is achieved by minimizing the total error between distance map obtained from both the dimension. This is given by

$$E = \frac{1}{\left(\sum_{i=1}^{N-1} \sum_{j=i+1}^N d_{ij}^*\right)} * \sum_{i=1}^{N-1} \sum_{j=i+1}^N \frac{(d_{ij}^* - d_{ij})^2}{d_{ij}^*}; \quad (1)$$

where N is number of nodes, d_{ij}^* is Euclidean distance between X_i and X_j in \mathbb{R}^p and d_{ij} is Euclidean distance between Y_i and Y_j in \mathbb{R}^d . In current 3-dimensional localization problem, $p = d = 3$.

We propose a modification to Sammon's error function so that it can handle the IDM which can be then used as a method to solve the localization without having to compute the FDM using algorithms like Dijkstra's. If there is no connection with the

nodes, then projection is omitted for that edge. The new error function is then defined as

$$E = \frac{1}{\left(\sum_{i=1}^{N-1} \sum_{j=i+1}^N d_{ij}^*\right)^2} * \sum_{i=1}^{N-1} \sum_{j=i+1}^N \frac{(a_{ij})(d_{ij}^* - d_{ij})^2}{d_{ij}^*}, a_{ij} = \begin{cases} 1 & \text{if edge } ij \text{ exists} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

This error can be now used as cost/fitness function for minimization employing various methods. Similar weighting approach to Sammon's stress has been previously successfully applied in data visualization [8] [9]. Using the above error function would provide us with a fast and cost effective solution with low mapping error. We have tried two gradient based line search methods (Sammon's original gradient descent approach) and two evolutionary methods (PSO and GA) as explained below.

1. Line search methods

We have tried out the gradient descent method proposed by [4] and Quasi-Newton line search method.

- **Gradient descent:** In this method a line search is made along the path of gradient descent [4]. Parameter, magic number, is kept at 0.3 for 500 iterations.
- **Quasi-Newton method:** In this method a line search is made, in the direction which is determined by hessian update. We have used *fminunc* provided by matlab. Except for number of iterations changed to 100, default parameters were used.

2. Metaheuristic Optimization Algorithms

Advantage of swarm/genetic methods is due to community approach, where each member tries to find the solution on their own using evolution and share the information with other members of the population. The solution thus found is expected to be a global minimum. These algorithms have been found to be robust even on the problems where dimensions explode.

- **Particle Swarm Optimization:** Developed by J. Kennedy and R. Eberhart [5] for optimizing fitness functions, is modeled on swarm behavior. In this method individual particle looks for the solution which is the best for given Sammon's fitness function. Each particle will update its search direction based on the local best, *PBest*, and the global best, *GBest* and the information is shared to the other particles in swarm. PSO has been used to solve the nonlinear mapping problem by Figueroa et al. [2]. We have used the standard PSO to minimize the error function defined in Eq. 2. Parameter ω is decremented over the for first 40 iterations between range (0.9,0.4) and for rest of iterations kept at constant at 0.4, $C_1 = C_2 = 2$, $V_{max} = 0.1$, $V_{min} = -0.1$, particle positions are normalized and limited to $X_{max} = 1$ and $X_{min} = 0$, respectively. Total number of iterations made is 2000. The swarm representing global best is selected and reshaped as the new co-ordinates of the sensor nodes.

- Genetic algorithm:** The solutions are found based on principles evolution of species, e.g., selection, recombination and mutations, has been used to minimize the fitness function Eq. [2]. Here, for the given localization problem we have used the evolutionary multidimensional scaling previously presented by Iswandy et al [7]. Genetic population which fits best for the localization problem is selected as the new co-ordinates of the sensor nodes. As parameters for 100 iterations, 20 number of populations were used, recombination rate was set at 0.1, mutation rate was kept at 0.9.

4 Simulations

For simulations, silhouette localization simulator developed by Whitehouse et al. [6] was modified for 3-dimensional scenarios. New localization algorithms based on Sammon’s non-linear mapping, PSO and genetic algorithms were added to the existing ones in the simulator. Assessment and visualization functions were also modified for 3-dimensional needs.

4.1 Experiments

Simulation process for localization algorithms involves *deployment*, *ranging*, *localizing* and *analysis* steps respectively.

In a cuboidal volume of [10m X 10m X 10m], 125 sensor nodes and 18 anchors/base stations were deployed. Sensors were deployed in random grid and anchors were deployed on uniform grid as shown in the Fig. [1(a)] below.

Based on the power capacity of sensor node, maximum range d_{max} which corresponds to maximum reachability the wireless signal is defined. Ranging process involves assigning of d_{max} , with additive Gaussian noise, to model connectivity between nodes in real deployment scenarios. This can be represented by noisy sphere model, with radius d_{max} and surface perturbed by Gaussian noise. Variations in

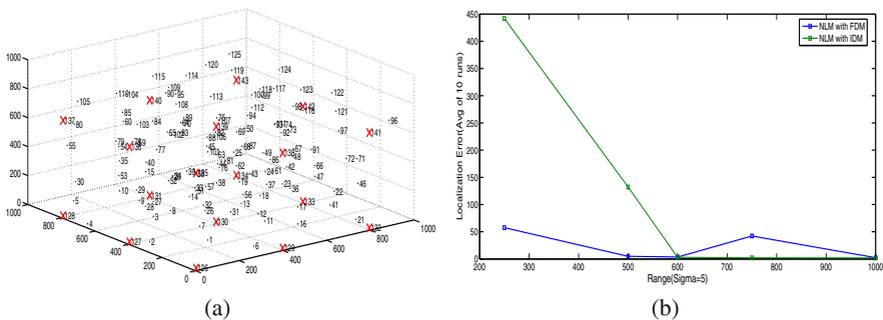


Fig. 1 Plots of experiments: (a) Deployment of anchors and sensors in 3D space; (b) Localization error vs. Range using NLM

range, d_{max} would lead to variations in connectivity between nodes providing us with different IDM. Raising the d_{max} increases connectivity among the nodes. Table 1 shows the average connectivity of nodes for different ranging levels. Average connectivity does not vary much with the variations in σ but is more dependent on d_{max} . For ranging, d_{max} is selected to be 2.5m, 5m, 6m, 7.5m and 10m. Gaussian noise levels(σ) was kept at levels 5, 7, 9, 10, 11, 13 and 15 respectively.

Together with the different noise levels σ and ranges d_{max} , 35 experiments in total were conducted. For each experiment, localization algorithms explained in above sections was executed 10 times with both IDM and FDM.

4.2 Results and Observations

For each run, *Sammon's stress* explained in Eq. 1 and *localization error* are observed. Localization error is calculated as the mean of distances between the computed and actual co-ordinates for all the nodes which are localized. For N nodes, this *LocError* is calculated as:

$$LocError = \frac{2}{N * (N - 1)} * \sum_{\forall i, i > j}^N \sqrt{(x_{ij}^* - x_{ij})^2 + (y_{ij}^* - y_{ij})^2 + (z_{ij}^* - z_{ij})^2} \quad (3)$$

where x_{ij}^* , y_{ij}^* , z_{ij}^* are the original co-ordinates and x_{ij} , y_{ij} , z_{ij} are calculated co-ordinates.

LocError is composed of all the three errors for FDM based localization as explained in Section 2. Here, for IDM based localization, *LocError* is only due to ranging errors and mapping error. Mean of 10 executions for each data set is shown in the graphs shown. Performance of both IDM and FDM based algorithms are depicted in Fig. 2 for the different values of range d_{max} with σ fixed at 5. NLM was found to provide better results as compared to other algorithms. From Fig. 2 it can be observed that as the connectivity between the nodes increases the localization error from IDM is less than the localization error of FDM.

NLM localization error variations with respect to range for noise level, $\sigma = 5$, for both IDM and FDM is shown in Fig. 1(b). Figure 3 presents variation of

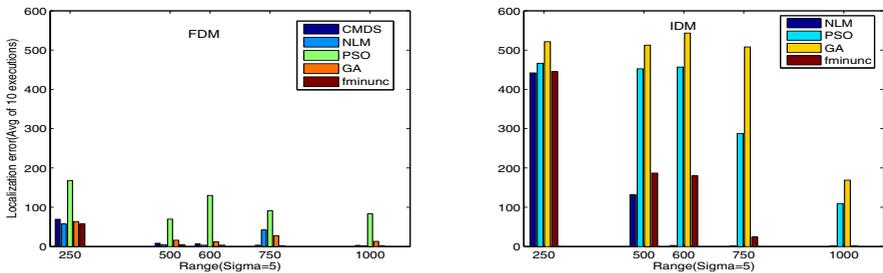


Fig. 2 Localization error vs. algorithms

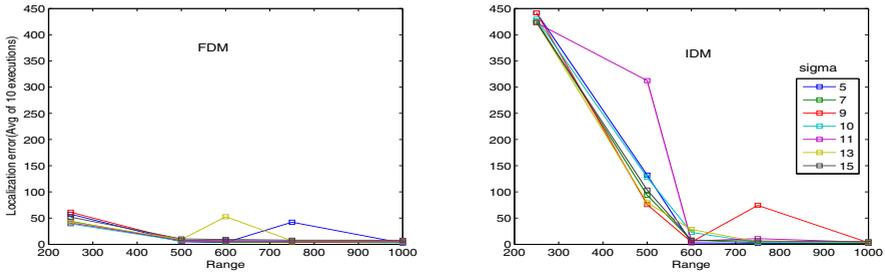


Fig. 3 Localization error vs. Range for different Noise levels using NLM

Table 1 Variations of connectivity and Errors with different range, d_{max}

Range	Connections	Connectivity in percent	Localization Err using FDM mean(std) ¹	Sammon's Err using FDM mean(std)	Localization using IDM mean(std)	Err Sammon's using IDM mean(std)
250	8	5.6	57.5 (0)	3.9E-02 (7.4E-15)	441.8 (27.9)	3.8E-01 (0.02)
500	46	32.2	4.0 (9.3E-16)	2.5E-04 (5.7E-20)	132.1 (92.2)	3.2E-02 (0.02)
600	67	46.8	3.4 (0)	7.3E-05 (1.4E-20)	2.7 (0.04)	6.08E-03 (0.01)
750	100	69.9	42.1 (126.2)	2.4E-02 (0.07)	1.9 (0.02)	6.1E-06 (1.4E-08)
1000	135	94.41	2.19 (4.6E-16)	1.10E-05 (1E-21)	1.68 (0.010)	4.7E-06 (1.7E-08)

¹ Standard deviation of 10 executions.

localization error with respect to range for each noise level, σ , for both IDM and FDM based NLM. It can be observed that as the range is increased the performance of algorithm with IDM matches to the level of FDM. This is due to increase in the connectivity information. From Table 1 it can be implied that localization based on IDM is more robust as the connectivity increases.

5 Conclusion

We have investigated an option to reduce computational effort in 3D localization in WSN by employing sparse distance information in a modified cost function in Sammon's Mapping. We have successfully demonstrated that localization for WSN in 3D scenarios can be achieved with the help of sparse distance matrix. From these simulations it can be observed, that IDM is superior until reaching 32 percent of average connectivity. For lower connectivities, a trade-off between localization accuracy and computational complexity or effort is required. In future improvements could be achieved by setting missing distance to the range d_{max} and use real valued weights from [0,1] in the modified cost function. In particular, when the sensor nodes are mobile, localization based on IDM is most promising with regard to computationally cost effective solution for moderate connectivity.

Our second goal, to improve the mapping error itself by using PSO and GA for the FDM as well as IDM case could not fully be reached. The achieved results

are not yet competitive to those of gradient descent techniques (see Fig. 2). Possibly high sensitivity of evolutionary techniques to the parameter settings cause this unexpected behavior. Optimizing PSO and GA by tuning the parameters will be investigated in our future work. Application of localization using IDM will be evaluated for patch and stitch methods available in localization. As the connectivity is much higher, when the small regions of clusters containing few nodes are formed, performance improvement in patch and stitch methods of localization is expected.

Acknowledgements. The work is in part supported by the Federal Ministry of Education and Research (BMBF) in the program AVS-mst, project PAC4PT-ROSIG, grant no. 16SV3604.

References

1. Chintalapudi, K., Govindan, R., Sukhatme, G., Dhariwal, A.: Website: Ad-Hoc Localization Using Ranging and Sectoring. In: INFOCOM (2004)
2. Figueroa, C.J., Estevez, P.A., Hernandez, R.E.: Nonlinear mappings based on particle swarm optimization. In: Proceedings. 2005 IEEE Int. Joint Conf. on Neural Networks, IJCNN 2005, July 31-August 4, vol. 3, pp. 1487–1492 (2005)
3. Dijkstra, E.W.: A note on two problems in connexion with graphs. *Numerische Mathematik* 1(1), 269–271 (1959)
4. Sammon, J.W.: A nonlinear mapping for data structure analysis. *IEEE Trans. on Computers* C-18(5), 401–409 (1969)
5. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proc. 1995 IEEE Int. Conf. on Neural Networks IJCNN (1995)
6. Whitehouse, K., Culler, D.E.: A robustness analysis of multi-hop ranging-based localization approximations. In: Proc. of the 5th Int. Conf. on Information Processing in Sensor Networks, IPSN 2006 (2006)
7. Iswandy, K., König, A.: Evolutionary Multidimensional Scaling for Data Visualization and Classification. In: Tiwari, et al. (eds.) *App. of Soft Computing: Recent Trends*, May 18, 2006, pp. 177–186. Springer, Heidelberg (2006), ISBN: 3-540-29123-7
8. König, A.: Dimensionality Reduction Techniques for Interactive Visualisation, Exploratory Data Analysis, and Classification. In: Pal, N.R. (ed.) *Pattern Recognition in Soft Computing Paradigm*. FLSI Soft Computing Series, vol. 2, pp. 1–37. World Scientific, Singapore (2001)
9. König, A.: Interactive Visualisation and Analysis of Hierarchical Neural Projections for Data Mining. *IEEE Transactions on Neural Networks TNN*, Special Issue on Neural Networks for Data Mining and Knowledge Discovery 11(3), 615–624 (2000)
10. Shang, Y., Ruml, W., Zhang, Y., Fromherz, M.P.J.: Localization from mere connectivity. In: Proc. of the 4th ACM Int. Sym. on Mobile ad hoc networking and computing MobiHoc (2003)
11. Ramadurai, V., Sichitiu, M.L.: Localization in Wireless Sensor Networks: A Probabilistic Approach. In: ICWN 2003 (2003)

Classification of Pressure Drop Devices of Proto Type Fast Breeder Reactor through Seven Layered Feed Forward Neural Network

Ramadevi Rathinasabapathy, Sheela Rani Balasubramaniam,
Manoharan Narayanasamy, Prakash Vasudevan, Kalyasundaram Perumal,
and Baldev Raj

Abstract. This paper presents a method to analyze the quality of pressure drop devices used for flow zoning in Prototype Fast Breeder Reactor (PFBR) by analyzing the occurrence of cavitations. In this work artificial neural network (ANN) has been used to classify the pressure drop devices as cavitating or not cavitating, under given operating conditions. A multi layer feed forward network with resilient back propagation algorithm has been used. The magnitude of root mean square (RMS) of the time signal acquired from an accelerometer installed downstream of various flow zones (totally 15) are fed as feature to the network for training and testing. Once adequately trained, the Neural Network based cavitation detection system would serve as an automated scheme for predicting the incipient cavitation regime and cavitation characteristics of a pressure drop device for a particular flow zone.

1 Introduction

Fast Breeder Reactors form a large energy resource. A PFBR comprises of primary circuit housed in reactor assembly and secondary sodium circuit as shown in

Ramadevi Rathinasabapathy · Sheela Rani Balasubramaniam · Manoharan Narayanasamy
Sathyabama University, Jeppiaar Nagar,
Rajiv Gandhi Road, Chennai 600119
e-mail: rama_adarsh@rediffmail.com, kavi_sheela@yahoo.com,
indiamano123@yahoo.com

Prakash Vasudevan · Kalyasundaram Perumal · Baldev Raj
Indira Gandhi Centre for Atomic Research, Kalpakkam,
Chennai 603102
e-mail: parkash@igcar.gov.in, pkalyanasundaram@igcar.gov.in,
dir@igcar.gov.in

fig 1. The reactor core consists of 1758 subassemblies including 181 fuel sub - assemblies [1]. The core has been divided into 15 flow zones to regulate flow in proportion to the heat generated in the subassembly. This is achieved by installing pressure drop devices at the foot of the subassembly [2]. These devices should meet the pressure drop requirements without any cavitation. Thus cavitation free performance of the device must be ensured. The paper is organized as follows, Section 2 reviews the cavitation phenomenon, section 3 describes data acquisition module, section 4 describes ANN modeling module for classification, section 5 analyses the results and performance and section 6 describes conclusion and future work.

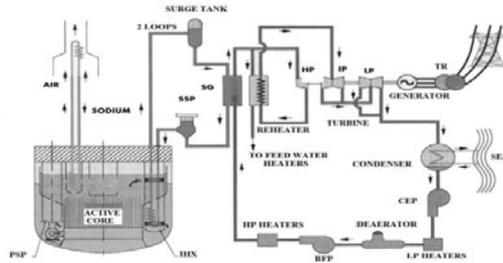


Fig. 1 PFBR flow sheet

2 Review on Cavitation Phenomenon

Cavitation is a general term used to describe the behavior of voids or bubbles in a liquid. Cavitation can arise in hydrodynamic flows when the pressure drops. This effect is, however regarded to be a destructive phenomenon for the most part. The cavities are formed into a liquid when the static pressure of the liquid for one reason or another is reduced below the vapour pressure of the liquid in current temperature. When cavities are carried to higher pressure region they implode violently and very high pressures can occur. When the local pressure of a liquid is reduced sufficiently, dissolved air starts to come out of liquid. In this process, air diffuses through wall into cavity. When pressure in the liquid is further reduced, evaporation pressure of the liquid is achieved. At this point the liquid starts to evaporate and cavities start to fill with vapour. When this kind of a cavity is subjected to a pressure rise cavity growth is stopped and once the pressure gets higher, cavities start to diminish [3].

Cavitation may appear to be a problem in fluid power systems. Cavitation affects fluid power systems and components in various ways, which are usually undesirable. For example efficiency of a system is reduced; a vibration as well as noise level of a system is increased due to cavitation. One of the remarkable

consequences of cavitation is cavitation erosion. Cavitation erosion is formed when cavitation is violent enough and occurs close to adjacent material surfaces. Cavitation erosion is always very harmful as it causes fluid contamination, leakage, blockages and undesired operation of system. Cavitation can be caused by vaporization, air ingestion, internal recirculation, flow turbulence, vane passing syndrome etc. When actions for preventing cavitation are considered, it is essential to verify the existence of cavitation and locate the cavitation inception point.

3 Experimental Setup and Instrumentation Employed for Data Acquisition

The cavitation testing is conducted in a water test loop. An assembly of orifice plate is fixed in the subassembly foot to simulate the upstream conditions. Based on velocity similitude required, water flow at about 40°C is sent through the test section. The flow is maintained constant by controlling the discharge and pump by pass valve in the test loop.

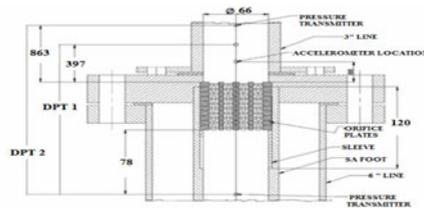


Fig. 2 Cavitation Test Loop

Fig. 2 shows blown out view of the instrumented test section with the downstream accelerometer installed to pick the acoustic noise. Cavitation is induced and acoustic noise is recorded by a high frequency accelerometer (resonant frequency: 50 kHz) installed downstream of the test section. The acoustic signal from the accelerometer is fed to an amplifier and filtered [5 kHz High pass] to avoid low frequency background vibration [4]. Acoustic signal for two sets of flow rates viz., 100% and 110% are recorded. Flow rate is measured by volume collection method with an estimated accuracy of $\pm 0.5\%$.

4 ANN Modeling Module

4.1 Network Architecture

One of the most commonly used networks is the multi layer feed forward network. Feed-forward networks are advantageous as they are the fastest models to execute.

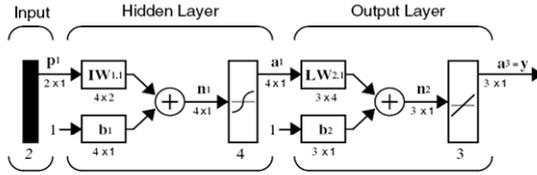


Fig. 3 Architecture of a feed forward network

In a feed forward network, information flows in one direction along connecting pathways, from the input layer via the hidden layers to the output layer. There is no feedback. That is, the output of any layer does not affect that same or preceding layer. It performs a weighted sum of its inputs and calculates an output using certain predefined activation functions. Activation functions for the hidden units are needed to introduce the nonlinearity into the network. The number of neurons and the way by which the neurons are interconnected defines the neural system architecture. The network is fed with a set of input-output pairs and trained to reproduce the outputs. Network structures, number of layers, and number of neurons in each layer, transfer function in each layer, define network specifications. To find the optimum network for detecting cavitation stages, we have been analyzing the above said parameters.

4.2 Resilient Back Propagation Algorithm

During training the weights and biases of the network are iteratively adjusted to minimize the network performance function. The default performance function for feed forward networks is mean square error (MSE) the average squared error between the network outputs and the target outputs is used. Resilient Back Propagation (trainrp) algorithm was used in the present work. Multi layer networks typically use sigmoid transfer functions in the hidden layers. They compress an infinite input range into a finite output range. Sigmoid functions are characterized by the fact that their slopes must approach zero, as the input gets large. This causes a problem when using steepest descent to train a multi layer network with sigmoid functions, because the gradient can have a very small magnitude and, therefore, cause small changes in the weights and biases, even though the weights and biases are far from their optimal values. The purpose of the Resilient back propagation training algorithm (RPROP) is to eliminate these harmful effects of the magnitudes of the partial derivatives. The size of the weight updates determine by individual update value Δ_{ij} .

$$\Delta ij^{(t)} = \begin{cases} \eta^+ * \Delta ij^{(t-1)} , & \text{if } \frac{\partial E^{(t-1)}}{\partial wij} * \frac{\partial E^{(t)}}{\partial wij} > 0 \\ \eta^- * \Delta ij^{(t-1)} , & \text{if } \frac{\partial E^{(t-1)}}{\partial wij} * \frac{\partial E^{(t)}}{\partial wij} < 0 \\ \Delta ij^{(t-1)} , & \text{else} \end{cases} \quad (1)$$

Where $0 < \eta^- < 1 < \eta^+$

Only the sign of the derivative is used to determine the direction of the weight update; the magnitude of the derivative has no effect on the weight update. The size of the weight change is determined by a separate update value. The update value for each weight and bias is increased by a factor delt_inc , decreased by a factor delt_dec and if the derivative is zero, then the update value remains the same. Whenever the weights are oscillating, the weight change is reduced. If the weight continues to change in the same direction for several iterations, then the magnitude of the weight change increases [5].

Resilient Back propagation Algorithm is generally much faster than the standard steepest descent algorithm. It also has a very good feature that it requires only a modest increase in memory requirements.

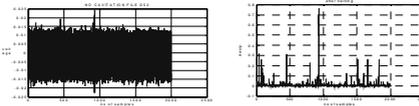
5 Performance Analysis

In the present work the optimum network architecture evolved out through an elaborate trial and error procedure. The best trained feed forward network containing seven layers. The transfer functions used for those layers are tansig (for input), logsig (for all hidden layers) and purelin (for output). The number of neurons used in each hidden layer is 50, 40, 30, 20, 15 and 10 respectively. TRAINRP was used with Learning rate = 0.01; Momentum constant = 0.9; Minimum performance gradient = $1e-10$; as training algorithm. For each zone separate goals were fixed and network was trained with 15 files (each file containing 2002 samples) and the rest of the files were given for testing. Four data sets have been analyzed viz zone II, Zone IV, Zone VI and Zone VII. Zone II contains 58 files for channel 1, Zone IV contains 78 files of both channel 1 and channel 2, Zone VI has 28 files but 15 files has both channel 1 and channel 2 and 13 files has only channel 1 and Zone VII has 68 files containing both channel 1 and channel 2.

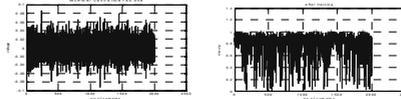
The first major component of this analysis involved evaluating the dependence of the cavitation signal on the magnitude of the signal and cavitation stage. Figure 5 represents the criteria used for cavitation conditions based on root mean square value of acoustic time signal, which indicate the characteristics of

cavitation formation under specified conditions. From figure, it is observed that the magnitude increases with the development of cavitation. There is a small rise at the incipient stage, 3 – 4 times rise in the well-developed stage. So magnitude has been chosen as a feature input to neural network. Initial processing of signal is carried out on neural network and through vigorous analysis of various cavitation signals; the classification range has been obtained from simulated output. The classification range has been fixed as, for no cavitation -0.009 to 0.09 , incipient cavitation 0.1 to 0.99 and for developed cavitation 1 to 2.9 . Following plots shows the original signal and simulated (after initial processing) signal.

No Cavitation:



Incipient Cavitation:



Developed Cavitation:

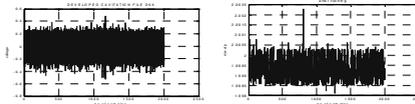


Fig. 4 Plots of original and simulated network signal

Table 1 gives the range selection for classification of cavitation based on the simulated network range.

Table 1 Range selection for classification of cavitation

ZoneVI Ch 1	Original Signal	Simulated Signal	Classification
NC File	-0.0227 to 0.024	-0.0044 to 0.7519	-0.009 to 0.09
IC File	-0.0923 to 0.0860	-0.0036 to 0.9994	0.1 to 0.99
DC File	-0.5270 to 0.4930	1.9981 to 2.0023	1 to 2.9
IC Towards NC	-0.0545 to 0.0600	-0.0070 to 0.9980	-
IC Towards DC	-0.1590 to 0.1580	0.9958 to 1.6890	-

Table 2 justifies the choice of selecting more number of layers. Developed Cavitation from Zone II Channel 1(n55) is analyzed for choice on number of layers and neurons. The network was trained and tested and the following results were obtained.

Table 2 Choice on number of layers & hidden neurons

Layers	Network	NC	IC	DC	Detection %
2	[10,1]	0	965	963	48.1
3	[15,10,1]	0	661	1265	63.1
4	[20,15,10,1]	0	552	1047	52.2
5	[30,20,15,10,1]	0	428	1264	63.1
6	[40,30,20,15,10,1]	0	103	1108	55.3
<u>7</u>	<u>[50,40,30,20,15,10,1]</u>	<u>0</u>	<u>310</u>	<u>1450</u>	<u>72.4</u>
8	[60,50,40,30,20,15,10,1]	0	541	1209	60.3

Table 3 shows performance analysis of seven layered feed forward neural network, with amplitude as input and trained using resilient back propagation algorithm. The efficiency of the network has been tested on all zones. The analysis is carried out zone wise.

Table 3 Performance Analysis

ZONE	CHANNEL	Train Data	Test Data
II	1	73.6%	85.7%
IV	1	89.2%	68.5%
	2	80.7%	64.2%
VI	1	86.4%	85.6%
	2	-	100%
VII	1	96%	83.4%
	2	73.2%	61.6%

6 Conclusion and Future Work

The above analysis concludes that the combination of seven layers with 50, 40, 30, 20, 15, 10 as number of hidden neurons and the combination of activation function

Tansig (input), Logsig (hidden layers), Purelin (output) with Mean Squared Error (MSE) as Performance Function has been determined as the best feed forward neural network parameters for classifying the pressure drop devices as cavitating or noncavitating one. The over all percentage of cavitation detection of train data was found to be 83.18% and for test data 78.42%.

In Future work, rather than considering only amplitude as input to the network, features like signal frequency, power spectrum (power spectrum of a signal represents the contribution of every frequency of the spectrum to the power of the overall signal which is useful in many signal processing applications), skew ness, Kurtosis, RMS value, mean value and average of the signals can be extracted and applied as input.

References

- [1] Description of PFBR, <http://www.igcar.ernet.in/igc.2004/reg/neg/smspdfsDescription>
- [2] Prototype Fast Breeder Reactor –Preliminary Safety Analysis report, Chapter 5.2 Core Engineering (February 2004)
- [3] Koivula, T.: On Cavitation in Fluid Power. In: Proc. of Ist FPNI-PhD Symp., Hamburg, pp. 371–382 (2000)
- [4] Gupta, P.K., Kumar, P.A., Kaul, A., Pandey, G.K., Padmakumar, G., Prakash, V., Anandbabu, C.: Neural Network Based Methodology for Cavitation Detection in Pressure Dropping Devices of PFBR. In: Proc. of national Seminar on Non destructive Evaluation, Hyderabad, December 7-9 (2006)
- [5] Rajesh, R., Chattopadhyay, S., Kundu, M.: Prediction of Equilibrium Solubility of Co₂ in Aqueous Alkanolamines Through Artificial Neural Network. In: Chemeca 2006, September 17-20 (2006)
- [6] Neural Network Toolbox, Signal Processing Toolbox, Statistics Toolbox of MATLAB
- [7] Kumarci, K., Abdollahian, M., Kumarc, F., Dehkordy, P.K.: Calculation of Natural Frequency of Arch Shape using Neural Network. In: CSCE 2008 Annual Conference, SCGC, June 10-13 (2008)

Music Visualization by Means of Comb Filter and Relaxation Time According to Human Perception

Mahdieh Poostchi, Iman Kamkar, and Javad Mohebbi

Abstract. The visualization of an audio signal's time structure is the primary step for every rhythm synchronizer system. To achieve this, principle components of the musical signal beats as strength, tempo and onset time must be detected. A filter bank of 180 different comb filters (from 60 bpm to 240 bpm) is convolved with different styles of musical signal to visualize according to human perception. In each interval, tempo in frequency domain and then its onset time in time domain are extracted. Now the key point is how to cover the gap between each two adjacent beat occurrences, from the last beat terminal to the next beat onset time. A magic number of 100 milliseconds (corresponding to human perception) is used. This interval dichotomizes equally. The first 50ms is used to stretch the signal to the relaxation point at the zero and the next one to grow it up to the next consequent beat onset. Therefore, a continuous sinusoidal signal is presented to visualize the corresponding music rhythm. Different simulation plots of various music styles and their overlaps have illustrated the proposed method.

1 Introduction

"Music is Nothing but Liquid Architecture," DJ Spooky. To have this architecture, finding its components are essential. In fact everywhere that sound waves or air vibrations in different sequences and amplitudes have been interpreted as harmonies, music exists. These sounds have four prime attributes which effect their

Mahdieh Poostchi

Computer Engineering Department, Iran University of Science and Technology, Tehran, Iran

e-mail: ma_poostchi@comp.iust.ac.ir

Iman Kamkar

Department of Computer Science, Islamic Azad University, Mashhad Branch, Iran

e-mail: kamkar.iman@gmail.com

Javad Mohebbi

Department of Computer Science, Islamic Azad University, Quchan Branch, Iran

e-mail: javad.mohebi@gmail.com

Table 1 Musical sound attributes

Physical	Amplitude	Frequency	Spectrum	Duration
Perceptual	Loudness	Pitch	Timbre	Length

perceptions (table 1). First is the amplitude or the size of vibrations that perceived as loudness. Second is the pitch or the frequency of vibrations; third is the timbre or The shape of the frequency spectrum of the sound, and the last one is duration or time length in which note sounds [2, 3]. To visualize music, knowing the rhythmic concept corresponding to these attributes is required. This concept can be understood as foot-tapping or beat-tracking. Human track music's rhythm by comprehending its main beat. Beat detection involves exploring the fundamental frequency (tempo), calculating its energy (strength), and locating its occurrence time (beat onset). Having these three components, music's rhythm can be visualized easily. According to literatures there are five different categories to detect beats of a piece of music as rule based method, auto-correlative methods, oscillating-filters, histogram, multi-agent and probabilistic techniques [4]. Each has its own merits and demerits. Among all of these, oscillating filters act as well as human perceives rhythms. It has an explicit implementation and is a good method for causal analysis [4]. From the all filterbank oscillators, just one lines up with original musical signal. This filter is the most compatible one with the beat tempo. Then the phase of the oscillator will be the beat resonate location [5-7, 10, 11]. In this paper a filter bank of 180 comb filters is convolved with the musical signal. Each filter is a train of impulses. The period of the impulses resonate from 60 bpm (Lento-slowest tempo perceived by human) to 240 bpm (Preto limit) [8, 9]. The correlation between the train's impulses which has the closest similarity to the signal's tempo will have the maximum energy and will be selected. On the other hand, phase of the selected filter will be the beat onset time, in time domain. After determining 3 components, in this paper a sinusoidal continuous signal is designed to connect occurrences of two different adjacent beats. The signal returns back by a magic number of 100ms. Henceforth, the first 50ms is used to reach the signal to the relaxation point at the zero on the vertical axis and the next one grows it up to the next beat resonate location. Depicting the techniques, in section 2 the comb filter method is deliberated. Section 3 discusses how to cover the gaps. Section 4 includes the results and finally we have conclusion and future works in section 5.

2 Beat Detection (Rhythm Tracking)

2.1 Tempo Extraction

Tempo which is calculated as beats per minutes describes the speed of a piece of music as well as human percepts speed of a song. To have this feature, acquiring

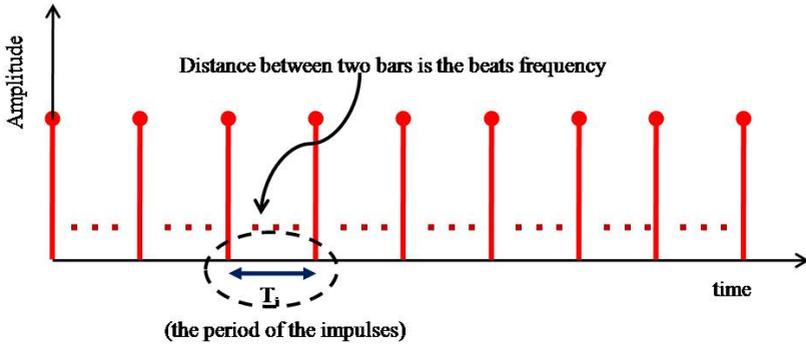


Fig. 1 Comb filter as a train of impulses

dominant frequency of the signal is inevitable [10-13]. Using comb filter, the rate of beats per minutes will be sought. Bank of comb filters contains 180 different filters of train impulses with periodicities T_i which is depicted in figure 1. The period of impulses T_i is calculated for different tempos as equation 1 defines.

$$T_i = \frac{tempo}{60} F_s \tag{1}$$

where tempo comes from the human perceptible tempos range, interval [60, 240] bpm, and F_s is the frequency sampling of the original signal. Table 2 includes the range of tempos according to *Westergaard*.

Table 2 The Range of Useful Tempos (Westergaard 1975, p. 274)

Beats/min	IOI(ms)	Tempo comment
30	2000	too slow to be useful
42	1414	Very slow
60	1000	moderately slow
80	700	Moderate
120	500	moderately fast
168	350	Very fast
240	250	too fast to be useful

After preparing the original signal to be convolved with these filters [5, 7], dominant periodicities are accumulated into a beat histogram [11]. This histogram contains a hierarchical model of a piece of music’s principle tempos and their relative strength. Therefore, peak of the histogram is the beat tempo of the piece.

2.2 Beat Strength

Beat strength enables us to distinguish two pieces of music with the same tempo. By this feature, for example one can say that a piece of Rock has higher amplitude (beat strength) than a piece of Classic music at the same tempo [3, 5]. By means of the beat histogram from the last step, the strength of a specific tempo can be determined easily from the vertical axis of the histogram.

2.3 Phase Detection

"Rhythmic response is crucially a phase phenomenon," *Scheirer*. One cannot tap against beat, or a bit ahead of or behind the beat, even if the tempo of tapping is accurate. It must occur exactly where the downbeat of the rhythm occurs. Finding this location is called onset or phase detection. By means of a comb filter which lines up with the signal tempo, finding downbeat location is practical in time domain. It needs to be convolved with the first period of the piece. Convolution is done by multiplication of Fourier transforms of the original signal and comb filter.

3 Covering Discontinuity

Principle components of a piece of music can shape a sinusoidal function as:

$$S = beat_i \sin(tempo_i / 60 \times (2\pi) + onset_i) \quad (2)$$

where $beat_i$ is the beat strength, $tempo_i$ is the fundamental frequency and $onset_i$ is the phase of the beat in the prospected interval. To make it clear, principle components of a piece of classical music (fig. 2) is extracted and is shown in fig. 3.

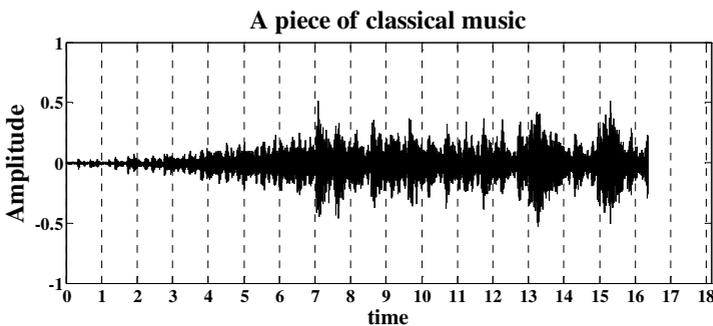


Fig. 2 Original signal of a piece of classical music. The beats onset time is shown by dashed lines every second.

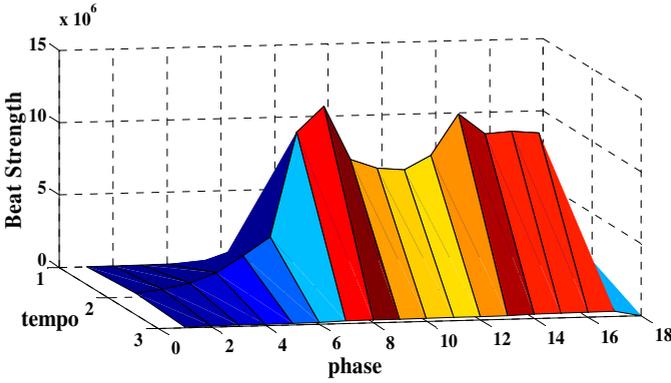


Fig. 3 Discrete rhythm visualization for signal in Fig. 2 by means of its 3 extracted components (strength, tempo and its phase)

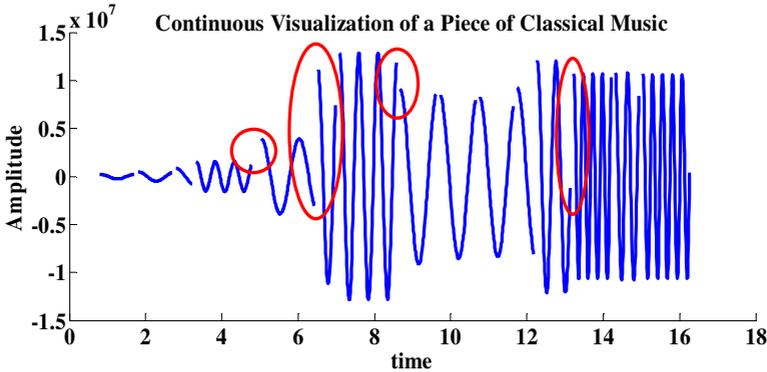


Fig. 4 Discrete simulated sinusoidal signal of fig. 2 (gaps can be seen every 1 sec(red circles))

These components formed the sinusoidal patterns in every second (fig. 4). As fig. 4 shows, to visualize a music rhythm there are gaps or discontinuities between each two adjacent beat occurrences. Connected bridges are required to cover the gaps, and to construct them a sufficient time interval is essential. How short the time between the onsets of successive events (inter-onset intervals/IOI) can be? In the range of 50ms to 100ms, one can hear individuated events. However, there is no rhythmic definition. In time perception, it has been demonstrated that if the time between events is too short ($<100\text{ms}$), listeners hear the sequences as a single, continuous event. But with 100ms threshold, we are able to hear individual event onset, and discern rhythmic shapes [14-16]. Hence the time 100ms is the magic number. The proposed method uses this magic number and dichotomizes it

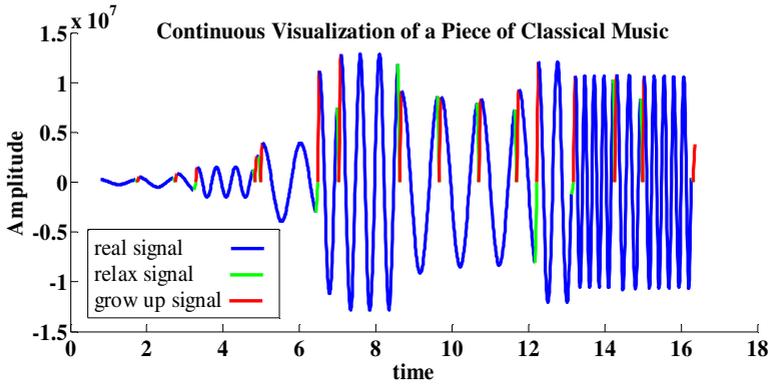


Fig. 5 Continuous simulation of classical piece of fig. 2 (by means of 100ms magic number as a connection bridge)

to two equal 50ms interval. In the gap between two adjacent events, first we return 100ms back from further event onset to the region of the previous event. Now we have a chance of 50 ms to relax our signal and stretch it from its position to the zero point at the amplitude axis (green lines). Henceforth, the next 50ms is used to reach our signal from the relaxation point to the next beat resonate location (red lines in fig. 5).

4 Results

A synchronizer system will act well if there is a pattern to show it what to do. Proposed algorithm in figure 6 gives the prospected pattern. Algorithm includes two main parts. In the first part the three principle components (beat strength, tempo and phase) must be extracted. In this paper a filter bank of 180 different tempos, from 60 bpm to 240 bpm has been used. In each considered interval, at first its fundamental periodicity has been calculated as beat's tempo. The energy of this tempo will be regarded as beat strength and at last, the phase of this resonate in the first period of considered interval is tempo's onset time. Initializing our algorithm, at first these features are obtained very quickly or if it is possible (the system is not real-time), we can just restore them. In second part, a continuous signal is formed. As is shown, this signal has three individual parts. Part i belongs to the original signal. It has an amplitude equal to the beat strength, beats, a frequency, ω , according to the original signal tempo (Hz), and a phase, φ , in beat resonance location. Part ii formed relaxed part of the signal. Here the signal will get relaxed and reached itself from anywhere to the zero point at amplitude axis by frequency 5 Hz. It is on account of that one quarter of our signal is just 50ms. Amplitude and

phase of this signal are equal to the amplitude and phase of the signal in part i with lag 100ms form next event. Finally, part iii grows up our simulated signal from its relaxation point, by frequency 5 Hz, to the next event onset. Different styles of music have been simulated by this algorithm. Besides what we have in last figures (excerpt from the second movement of Beethoven's 9th, as classical music), simulations for two other musical genres are presented in fig. 7. An excerpt of "Forever Be," by The Chemical Brothers, from techno and a piece of Jazz music (Descarga by Tito Puente). These plots - especially simulation plots in row 3- can play a big role as a synchronizer leader.

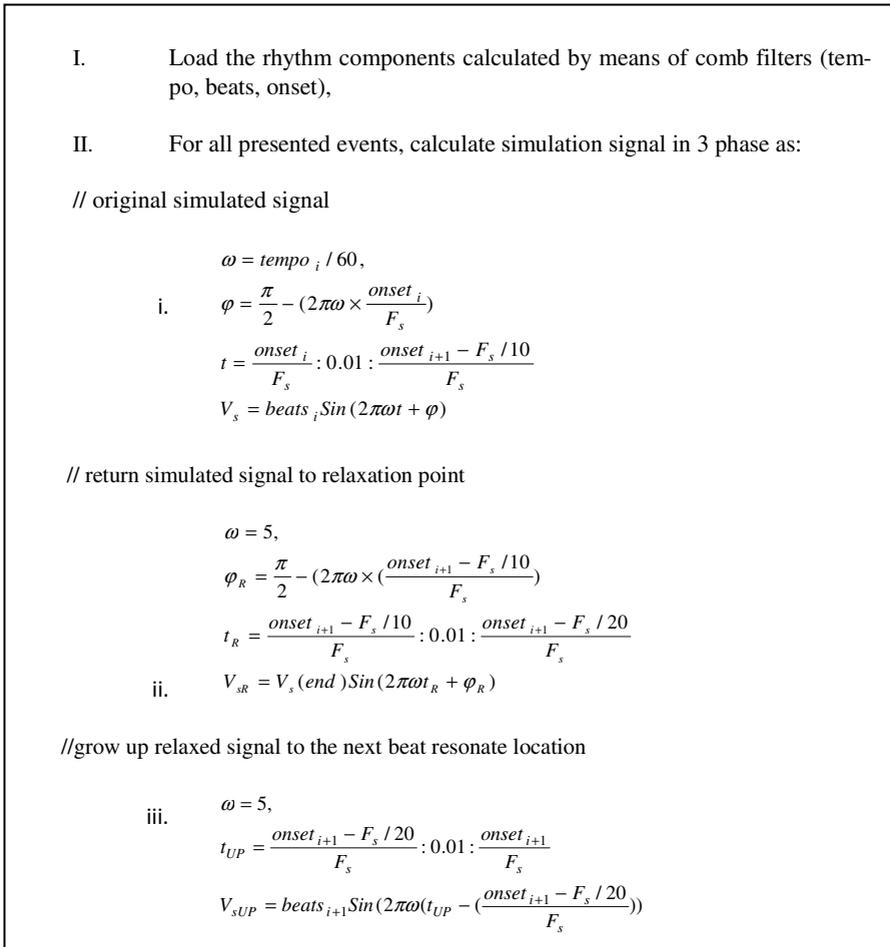


Fig. 6 Proposed algorithm to have a visualization of a piece of music

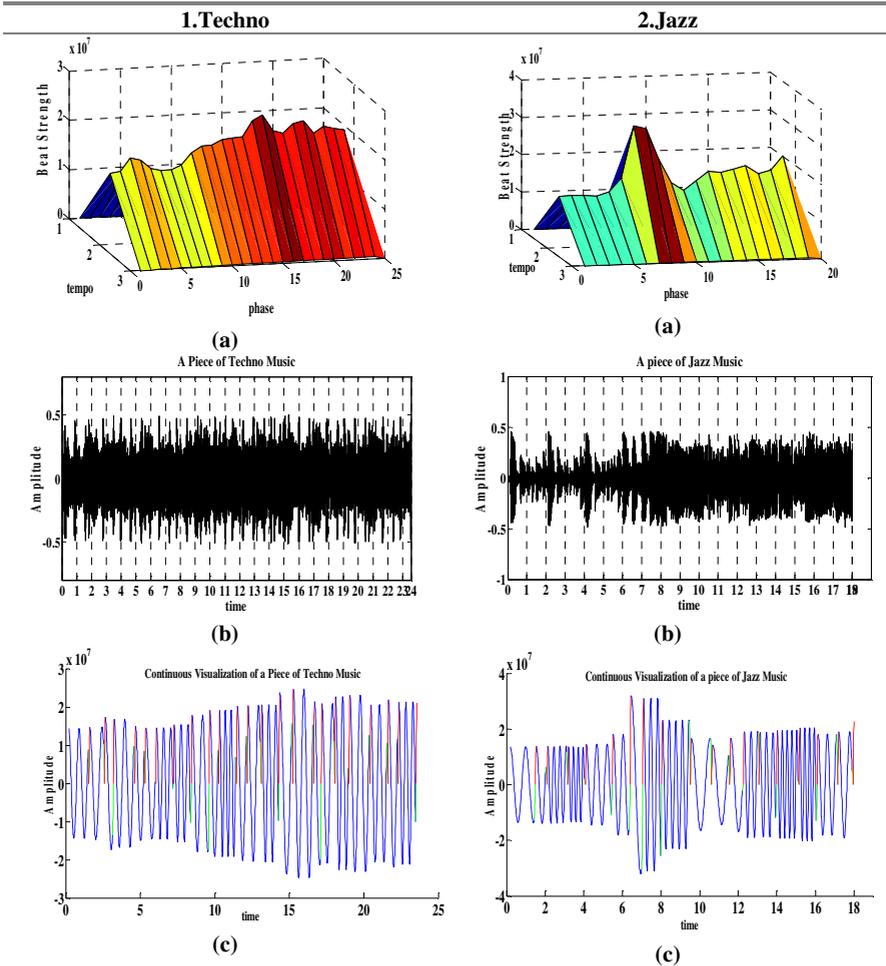


Fig. 7 1. An excerpt of Forever Be, by The Chemical Brothers, which classified as Techno, 2. A piece of Jazz music (Descarga by Tito Puente) from top to down: a) surface plot of 3 principle components (strength, tempo and phase), b) original signal, c) continuous simulation of original signal rhythm (blue line is the simulated signal, relaxed signal show in green and signal in red take relaxed signal to the next event resonance location).

5 Conclusion and Future Works

A dancer artificial muscle, a vibrator, an imitator doll, and so many other such systems, where the goal is to act like human when tap his foot or clap hands to show his perception of music, require a behavioral pattern. This paper gives such a

leader in form of a sinusoidal signal. Filtering the signal with 180 different comb filters is the most time consuming phase of this rhythm detection process.

When we act offline it can be ignored, but examine other methods such as using wavelet will improve both its speed and precision. It is obvious that separating signal spectrum to multiple individual envelopes where each envelope concerns to a certain frequency band will improve our detection accuracy. Human auditory system is more sensitive to low frequency than high frequency. Number of envelopes and the range of frequency they cover will have a direct effect on detection process. Regarding all of this, in this paper method outlined in *Scheirer* 1998, is used. A new separating method based on octave or logarithm band can also be beneficial to be developed.

References

- [1] Anne, L., Runehov, C.: Sacred Or Neural? In: The Potential of Neuroscience to Explain Religious Experience, Vandenhoeck & Ruprecht (2007), ISBN 3525569807, 9783525569801
- [2] Benson, D.: Music: A Mathematical Offering. Cambridge University Press, Cambridge (2007), ISBN: 0521853877
- [3] Kamien, R.: MUSIC An Appreciation, 5th edn. McGraw-Hill, Inc., New York (1992)
- [4] Hainsworth, S.W.: Techniques for the Automated Analysis of Musical Audio. Doctor of Philosophy thesis, University of Cambridge (December 2003)
- [5] Tzanetakis, G., Essl, G., Cook, P.: Human perception and computer extraction of beat strength. In: Proc. Digital Audio Effects Workshop (DAFx), pp. 257–261 (2002)
- [6] Tzanetakis, G., Essl, G., Cook, P.: Audio Analysis using the Discrete Wavelet Transform. In: Proc. Conf. in Acoustic and Music Theory Applications, WSES (September 2001)
- [7] Scheirer, E.: Tempo and beat analysis of acoustic musical signals. Journal of the Acoustical Society of America 103(1), 588–601 (1998)
- [8] Loundom, J.: Hearing Rhythmic Gestures: Moving Bodies and Embodied Minds. In: Keynote Address, Music and Gesture Conference, University of East Anglia, Norwich (August 2003)
- [9] Loundom, J.: Musical Rhythm: Motion, Pace and Gesture. In: Gritten, A., King, E. (eds.) Music and Gesture, pp. 126–141. Ashgate, Aldershot (2006)
- [10] Chang, Y.-Y., Lin, Y.-C.: Music Tempo (Speed) Classification, CS229 Autumn 2005 (2005)
- [11] Tzanetakis, G.: Tempo Extraction using Beat Histograms. In: Proceedings of the 1th Music Information Retrieval Evaluation eXchange, MIREX 2005 (2005)
- [12] Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. IEEE Trans. Speech and Audio Processing 10(5), 293–302 (2002)
- [13] Gold, B., Morgan, N.: Speech and Audio Signal Processing: Processing and Perception of Speech and Music. John Wiley and Sons, Chichester (2000)
- [14] Davies, J.B.: The psychology of music. Stanford University Press, Stanford (1978)
- [15] Krumhansl, C.L.: Rhythm and pitch in music cognition. Psychological Bulletin 126(1), 159–179 (2000)
- [16] Fraisse, P.: Rhythm and Tempo. In: Deutsch, D. (ed.) Psychology of Music, pp. 149–180. Academic Press, New York (1982)
- [17] Hong, S.-d., Park, J.-w., Lee, W.-H.: Designing Audio Visual Software for Digital Interactive Art. In: IEEE Hangzhou, pp. 651–655 (November 2006)

GA/SVM for Diagnosis Sleep Stages Using Non-linear and Spectral Features

Maryam Vatankhah, Mohammad Reza Akbarzadeh Totonchi,
Ali Moghimi, and Vahid Asadpour

Abstract. Human's sleep is divided into two segments, Rapid Eye Movement (REM) sleep and Non-REM (NREM) sleep. NREM sleep is further divided into 4 stages. Sleep staging attempts to identify these stages based on the signals collected in polysomnogram (PSG). Significant information can be derived from the EEG signals collected during PSG.

In our study we extract spectral features from EEG signals. Genetic Algorithm (GA) is used for selecting best features and then Support Vector Machine (SVM) with different kernels have been applied to differentiate sleep stages. According to chaotic characteristic of EEG signal, we use non-linear features, as well. Non-linear and spectral features can differentiate stages awake and REM with 98.15% accuracy. Chaotic features improved classification rate as a necessary component. Succinctly, through the feature space constructed by approximate entropy and fractal dimension, different stages of EEG signals can be recognized from each other expressly. That is to say, Pattern varies under the different sleep stages. Therefore Healthy humans with a regular night's sleep will follow these sleep stages in a particular pattern.

Maryam Vatankhah

Islamic Azad University of Mashhad, young researcher club members
e-mail: vatankhahmaryam@gmail.com

Mohammad Reza Akbarzadeh Totonchi

Associate Professor, Departments of Electrical and Computer Engineering,
Ferdowsi University of Mashhad
e-mail: akbarzadeh@ieee.org

Ali Moghimi

Associate Professor, Dept. of Biology, Faculty of Sciences Ferdowsi University of
Mashhad, FUM
e-mail: moghimi@um.ac.ir

Vahid Asadpour

Assistant Professor, Electrical Engineering Department of Sadjad University
e-mail: asadpour@aut.ac.ir

1 Introduction

Sleep staging, performed in a test called polysomnogram (PSG), is a common and important procedure for the diagnosis of sleep disorders. PSG studies a series of biomedical signals ranging from heart beat, blood oxygen concentration, muscle movements, to brain activity. The most important signal and also the most difficult to analyze is the electroencephalogram (EEG), which shows the brain's activities [1,2].

Currently, trained technicians manually analyze the relevant biomedical signals to generate a sleep stage classification for every 30 seconds of data, called an epoch. While softwares have been implemented to provide additional information on each epoch of data, such as frequency content and peak-to-peak voltage, no algorithm have been commercially used for computerized sleep staging. This automation would drastically reduce the amount of manual tasks, thereby making the process more reliable and cost efficient.

Much research has been dedicated to PSG interpretation. In particular, many studies focus on the analysis of EEG signals. Human analysis follow a set of rules defined in a manual by Rechtschaffen and Kales [3]. These rules define frequency, amplitude, and contextual parameters. Therefore, EEG analysis algorithms must take into consideration both the time and the frequency domain information. One method suggested in research is using a time-frequency representation called spectrogram.

Time frequency representation of EEG has been used in numerous researches as a tool to understand EEG's behavior. Nayak[4] used time-frequency analysis to demonstrate alpha blocking by anesthesia. Van Hese[5] used scalograms to analyze the practicality of using wavelet analysis for sleep staging. Scheuer[6] used spectrograms to demonstrate certain transient activities that can predict epileptic attacks. In a review on spectral analysis of biomedical signals, Muthuswamy[7] indicated wavelet analysis as being one of the most useful utility.

In our study we extract spectral features. According to chaotic characteristic of EEG signal and for achieving more accuracy, we use non-linear features, entropy and fractal dimension. Calculation spectral and non-linear features made the feature space too large so GA is used for selecting best features and decreasing the feature's space. Then SVM classifiers with different kernels have been applied to differentiate sleep stages.

Non-linear and spectral features can differentiate stages awake and REM with 98.15% accuracy. It can differentiate stages II and III with 91.52% and stages III and IV with 90.63% accuracy. The GA/SVM algorithm select the best features and classify awake and REM stages with highly accurate rate.

2 Data Collection

2.1 Sleep Staging

Human's sleep is divided into two segments, Rapid Eye Movement (REM) sleep and Non-REM (NREM) sleep. REM sleep is best characterized by the occurrence

of dreams. During REM sleep, the person receives psychological rest and the brain actively reorganizes itself into a better state. NREM sleep provides the person with physiological rest and the brain's activities slow down. NREM sleep is further divided into 4 stages I, II, III, and IV. Therefore, common practice recognizes 5 sleep stages. Healthy humans with a regular night's sleep will follow these sleep stages in a particular pattern. Traditionally, to complete the set of stages, Awake and Movement Time (MT - Movement Time represents the periods of time when the PSG signals are obscured by body movement) are added. Different sleep stages are shown in figure 1.

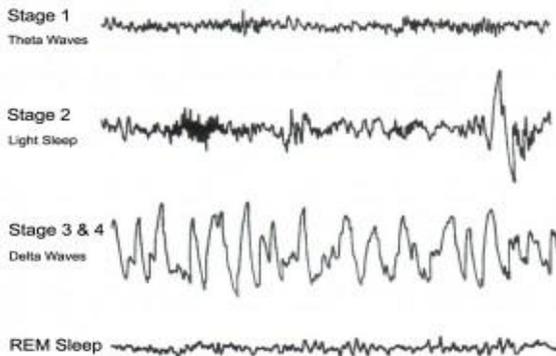


Fig. 1 different sleep stages

Sleep staging attempts to identify these stages based on the signals collected in PSG. Significant information can be derived from the EEG signals collected during PSG. EEG analysis looks primarily at the 6 key features, which are listed along with their characteristics below (Table 1).

Table 1 Frequency, characteristic of EEG features

feature	frequency	amplitude	temporal
Alpha Activity	8-13 HZ	20-60 μv	Present in awake, stage I and REM
Beta Activity	13+ HZ	2-20 μv	Dominant in awake
Theta Activity	4-8 HZ	50-75 μv	Present in stage I, II, III and IV
Delta Activity	0-4 HZ	75+ μv	Present in stage III and IV
Sleep spindles	12-14 HZ		Present in stage II

2.2 Data

This paper based on one set of 8 hour sleep study data provided by the Physio-Bank (physiologic signal archives for biomedical research). The PhysioBank stores the sleep study data in the European Data Format (EDF). The recordings were obtained from Caucasian males and females (21 - 35 years old) without any medication; they contain horizontal EOG, FpzCz and PzOz EEG, each sampled at 100 Hz. Hypnograms are manually scored according to Rechtschaffen & Kales based on Fpz-Cz / Pz-Oz EEG instead of C4-A1 / C3-A2 EEG [8].

After filtering the signals from 0.5 to 100 HZ, two dataset with 10 hour sleep records are used. Each of this data set contains 1200 epoch which evaluate with an expert. Table 2 has shown the number of epoch in two stages. Each set of epochs are stored in Matlab data files.

Table 2 The number of epochs in each stage

Sc4012e0	Sc4002e0	Data set/stage
176	255	Awake
92	59	I
660	373	II
80	94	III
16	203	IV
176	215	REM
0	1	MT

3 Feature Extraction

At first, Feature extraction contains 5 spectral features for each EEG dataset and 2 features for EOG datasets. Calculating non-linear features can improve the differentiability of sleep stages so entropy and fractal dimension for each EEG epoch are added to previous features.

In this section, the definition and characteristics of spectrograms and non-linear features are briefly introduced. Some of this information is extracted from [9].

3.1 Spectrogram

Spectrograms are basic TFR², where signals are transformed into the frequency domain in segments and realigned against time [10]. The segments of the signal are of equal length and they may overlap. Generally a windowing function $w(t)$ is used to isolate the segment. Each segment $s(t)$ is transformed by Short-Time Fourier Transform

² Time Frequency Representation.

$$STFT_X(t,f)=\int_{-\infty}^{+\infty} x(\tau)w^*(\tau-t)e^{-i2\pi f\tau} d\tau \quad (1)$$

Then the energy density spectrum is

$$SPEC(t,f)=|STFT_X(t,f)|^2 \quad (2)$$

And the result is called a spectrogram.

The spectrogram follows the Heisenberg Uncertainty Principle, meaning that it is impossible to have perfect resolution in both time and frequency. Spectrogram is time-dependent, meaning that the resolutions are dependent on the window size, in other words the length of signal segment. The wider the window, the better the frequency resolution and the worse the time resolution.

3.2 Mixed Frequency Detection

Mixed frequency activity is most predominant in awake stage. Since a natural transition from Awake is into Stage I, EEG segments from these two stages will be the samples studied to generate some mixed frequency features. The mixed frequency activity is most prominent in the spectrograms. Mixed frequency activity has two features, intense activity spread across low frequency ranges, and notable activity in the beta band appearing as slightly brighter spots. Therefore, the goal is to identify one or more features that can identify a spectrogram containing mixed frequency activity.

The following three features were considered.

1. Standard deviation across the entire frequency range: This feature is defined as

$$\text{std}_1=\text{std}\left(\frac{\int SPEC(t,f)dt}{\int tdt}\right) \quad (3)$$

Its applicability stems from higher ripples in the spectrogram of the awake stage. Since no particular time event need to be located, it is safe to average across time. In order to reduce the effect of averaging across positive and negative valleys, a second feature is calculated as the absolute value of the spectrogram, which is called std2.

2. Energy ratio between beta band and other lower frequency bands: This feature is determined as

$$\text{ratio} = \frac{\int_{13}^{40} SPEC(t,f)df}{\int_0^{13} SPEC(t,f)df} \quad (4)$$

This feature draws from the fact that the energy under the lower frequencies is high regardless of mixed frequency activities. However, the energy under beta band varies greatly depending the amount of mixed frequency. Upper frequency and ratio frequency features are used for EOG datasets [11].

$$\text{Upper Frequency}=\sum_t \sum_{f=4HZ}^{12HZ} SPEC(t,f) \quad (5)$$

$$\text{ratio Frequency}=\max\left(\frac{SPEC(t,4:12)}{SPEC(t,0:4)}\right) \quad (6)$$

3.3 Approximate Entropy

ApEn is a recently formulated family of parameters and statistics quantifying regularity (orderliness) in serial data. It has been used mainly in the analysis of heart rate variability [8-10], endocrine hormone release pulsatility [12], estimating regularity in epileptic seizure time series data [13] and estimating the depth of anesthesia [14]. ApEn assigns a non-negative number to a time series, with larger values corresponding to more complexity or irregularity in the data.

Because of the non-linear character of EEG signals, ApEn can be used as a powerful tool in the study of the EEG activity. Although m and r are critical in determining the outcome of ApEn, no guidelines exist for optimizing their values. In principle, the accuracy and confidence of the entropy estimate improve as the number of matches of length m and $m+1$ increases. For this study, ApEn is estimated with $m=3$ and $r=0.25$ times the SD of the original data sequence. By means of it, we are provided one dimensions for classification.

3.4 Fractal Dimension

Fractal dimension (FD) emphasizes the geometric property of basin of attraction. These dimension show geometrical property of attractors and is also computed very fast [15]. Our goal was to associate each 5-second segment data as a trial to its corresponding class. To do this, features were extracted from each 1-second segment with 50% overlap, and sequence of 9 extracted features was considered as the feature vector of a 5-second segment, which was to be modeled and classified. In Higuchi's algorithm, k new time series are constructed from the signal $x(1), x(2), \dots, x(N)$ under study:

$$x_m^k = \{x(m), x(m+k), x(m+2k), \dots, x(m + \lfloor \frac{N-m}{k} \rfloor k)\} \quad (7)$$

where $m = 1, 2, \dots, k$ and k indicate the initial time value, and the discrete time interval between points, respectively. For each of the k time series x_m^k , the length $L_m(k)$ is computed by:

$$L_m(k) = \frac{\sum_{i=1}^{\lfloor \frac{N-m}{k} \rfloor} |x(m+ik) - x(m+(i-1)k)| (N-1)}{\lfloor \frac{N-m}{k} \rfloor k} \quad (8)$$

Where N is the total length of the signal x . An average length is computed as the mean of the k lengths $L_m(k)$ (for $m = 1, 2, \dots, k$). This procedure is repeated for each k ranging from 1 to k_{\max} , obtaining an average length for each k . In the curve of $\ln(L(k))$ versus $\ln(1/k)$, the slope of the best fitted line to this curve is the estimate of the fractal dimension.

4 Gene Selection and Classification

We describe now the hybrid GA/SVM algorithm for carrying out gene selection and classification. The GA is designed for gene selection and The SVM-based classifier with different kernels is used to classify different sleep stages.

4.1 The Genetic Algorithm General Schema

Genetic algorithm (GA) is a biologically motivated adaptive system based on natural selection and genetic recombination. It utilizes selection, crossover and mutation mechanisms to evolve the population. The most important steps of this process is defined as follow[16]:

Step 1. (Constructing Fitness Function): The fitness function is the unique information while using Genetic algorithm to perform search, it plays a critical role to measure a algorithm's performance. When the fitting precision ε gets minimum, the fitness function is:

$$\varepsilon = |y_{\text{test}} - y_{\text{calculation}}| \quad (9)$$

Where y_{test} is the output of training samples, $y_{\text{calculation}}$ is the output of SVM model in which the c and τ have been predetermined.

Step 2. (Selecting strategy): A strategy based on fitness and elitist propagation is adopted. Assumed the fitness of single x_i is f_i , the possibility of x_i being selected p_{si} is:

$$\sum f_i p_{si} = \frac{f_i}{\sum f_i} \quad (10)$$

Where $\sum f_i$ denotes the sum of single-fitness in this group.

Step 3. (Crossover and mutation): Mutations are performed randomly by converting a "1" bit into a "0" bit or a "0" bit into a "1" bit. The rates of crossover and mutation are probabilistically determined.

Step 4. (Next generation): Form a population for the next generation.

Step 5. (Stop conditions): If the number of generations equals a given scale, then the best chromosomes are presented as a solution; otherwise go back to Step 2.

4.2 The SVM Classifier

Support Vector Machines are basically binary classification algorithms [17]. When the data are linearly separable, SVM computes the hyper plane that maximizes the margin between the training examples and the class boundary. When the data are not linearly separable, the examples are mapped to a high dimensional space where such a separating hyperplane can be found [18]. The mechanism that defines this mapping process is called the kernel function. SVM are powerful classifiers with good performance in the domain of EEG signals [19, 20]. In our wrapper GA/SVM algorithm, we use a SVM classifier to assess the quality of a gene subset. For a chromosome x that represents a gene subset, we apply a Cross-Validation method to calculate the average accuracy (rate of correct classification) of a SVM trained with this gene subset [21].

One of the key elements of a SVM classifier concerns the choice of its kernel. In our study, we have chosen to use the RBF kernel. We also experimented Gaussian, MLP and polynomial kernels. For polynomial kernels, the main difficulty is to determine an appropriate polynomial degree while the results we obtained with the Gaussian kernel are not satisfactory.

4.3 Parameter Setting

We extracted 18 features via non-linear and spectral methods. In order to decrease the feature space, GA was used. 7 features have been selected with GA and then SVM classifiers with different kernels have been applied.

For our GA/SVM algorithm, the GA is implemented in Matlab (Version 5.3.1 for Windows). The SVM classifier is based on the SVM Toolbox developed by Gavin Cawley³. The GA parameters used in our model of gene subset selection for the sleep datasets are shown in Tables 3. The normalization parameter C is fixed at 100 and the control parameter γ for the RBF kernel of SVM is fixed to 0.5.

Table 3 GA parameters

parameters	Sleep data
Size of population	50
Length of chromosome	100
Number of generations	50
Crossover rate	0.6
Mutation rate	0.01
Elitism rate	15%

5 Results

Table 4 summarizes our results for the sleep data sets with the different SVM kernels for awake and REM stages. It shows the RBF kernel has the best result.

Table 4 Comparison of GA/SVM with different kernels for awake and REM stages

Sleep Dataset features	GA&SVM kernels			
	RBF	Gaussian	Polynomial	MLP
Spectral features	92.31%	87.34%	89.8%	91.6%
Non-linear features	98.15%	90.87%	94.73%	95.7%

It can be observed, for classification REM and awake stages we obtain a classification rate of 98.15% with nonlinear and spectral features. We can differentiate stages II and III with 91.52% and stages III and IV with 90.63% accuracy.

The most interesting result we obtained with our model concern the sleep datasets is chaotic features makes the dataset more differentiable. Because of the non-linear character of EEG signals, non-linear features could improve the classification accuracy.

6 Conclusion

Spectrogram can differentiate Awake and REM based on the sleep staging manual's feature mixed frequency. The spectrogram features include standard deviation across frequency, energy ratio between beta activity and lower frequency activity, and edge detection in beta band were investigated. Non-linear and spectral features are shown to be the best in terms of performance dealing with Awake from REM differentiation, with 98.15% which is much better than reported in [22,23,24,25]. While differentiating stages II and III only has a performance of 91.52%, its differentiability from adjacent stages, Stages III and IV, is 90.63%. Studying spectrogram characteristics that identify Awake and REM derives the feature for mixed frequency activity.

In our study, non-linear features improved classification rate as a necessary component. Summarily, through the feature space constructed by ApEn and fractal dimension, different stages of EEG signals can be recognized from each other expressly. That is to say, pattern varies under the different sleep stages. Therefore Healthy humans with a regular night's sleep will follow these sleep stages in a particular pattern.

References

1. Parrino, L., Ferrillo, F., Smerieri, A., Spaggiari, M.C., Palomba, V., Rossi, M., Terzano, M.G.: Is insomnia a neurophysiological disorder? The role of sleep EEG microstructure. *Brain Research Bulletin* 63, 377–383 (2004)
2. Scher, M.S.: Automated EEG-sleep analyses and neonatal neurointensive care. *Sleep Medicine* 5, 533–540 (2004)
3. Sleep Computing of the Japanese Society of Sleep Researches(JSSR): Proposed supplements and amendments to 'A Manual of Standardized Terminology, Techniques and Scoring System for Sleep Stages of Human Subjects', the Rechtschaffen & Kales (1968) standard. *Psychiatry and Clinical Neurosciences* 55, 305–310 (2001)
4. Nayak, A., Roy, R.J.: Time Frequency Spectral Representation of EEG. *IEEE*, 7–8 (1993)
5. Van Hese, P., Philips, W., et al.: Automatic Detection of Sleep Stages using the EEG. In: Proc. of the 23rd Ann'l, EMBS Int'l. Conf., Istanbul, Turkey, October 25-28, pp. 1944–1947 (2001)
6. Scheuer, M.L.: Continuous EEG Monitoring in the Intensive Care Unit. *Epilepsia* 43(suppl. 3), 114–127 (2002)

7. Muthuswamy, J., Thakor, N.V.: Spectral analysis methods for neurological signals. *Journal of Neuroscience Methods* 83, 1–14 (1998)
8. <http://www.physionet.org/physiobank/database/sleep-edf/>
9. Laing, R.J.A.: A Sleep Spindle Detection Algorithm. Master Thesis
10. Shen, J.Y.: Sleep Staging: Study through Spectrograms and Scalograms (2003), <http://www.eng.uwaterloo.ca/~y5shen/research/sleepstaging.htm#sleepstage>
11. Shen, J.Y.: Applying Image Processing to Identify Characteristic Waves in EOG (2004), <http://www.eng.uwaterloo.ca/~y5shen/research/sleepstaging.htm#sleepstage>
12. Pincus, S.M.: Older males secrete luteinizing hormone and testosterone more irregularly and joint more asynchronously, than younger males. *Proc. Natl. Acad. Sci. USA* 93, 14100–14105 (1996)
13. Radhakrishnan, N., Gangadhar, E.N.: Estimating regularity in epileptic seizure time-series data. A complexity-measure approach. *IEEE Eng. Med. BIOI.* 17, 89–94 (1998)
14. Zhang, X., Roy, R.J.: Derived fuzzy knowledge model for estimating the depth of anesthesia. *IEEE Trans. Biomed. Eng.* 48 (2001)
15. Solhjo, S., Motie Nasrabadi, A.: EEG-Based Mental Task Classification in Hypnotized and Normal Subjects. In: *Engineering in Medicine and Biology 27th Annual Conference*, Shanghai, China, September 1-4 (2005)
16. Zhou, J., Bai, T.: The SVM Optimized by Culture Genetic Algorithm and Its Application in Forecasting Share Price. In: *IEEE International Conference on Granular Computing*, pp. 838–843 (2008)
17. Joachims, T.: Estimating the Generalization Performance of a SVM Efficiently. In: *Proceedings of the International Conference on Machine Learning (ICML)*. Morgan Kaufman, San Francisco (2000)
18. Huerta, E.B., Duval, B.: A Hybrid GA/SVM Approach for Gene Selection and Classification of Microarray Data, pp. 34–44. Springer, Heidelberg (2006)
19. Güler, I., Beyli, E.Ü.: Multiclass Support Vector Machines for EEG-Signals Classification. *IEEE Transactions on Information Technology in Biomedicine* 11(2) (March 2007)
20. Guyon, J.W., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. *Machine Learning* 46(1-3), 389–422 (2002)
21. Mukherjee, S.: *Classifying Microarray Data Using Support Vector Machines*. Springer, Heidelberg (2003)
22. Shimada, T., Shiina, T., Saito, Y.: Sleep Stage Diagnosis System with Neural Network Analysis. In: *Proc. of the 20th Ann'l. Int'l. Conf. of the IEEE Engineering in Medicine and Biology Society*, vol. 20(4), pp. 2074–2077 (1998)
23. Oropesa, E., Cycon, H.L., Jobert, M.: Sleep Staging Classification using Wavelet Transform and Neural Network. TR-99-008, International Computer Science Institute, Berkeley, California, (March 30, 1999)
24. Pohl, V., Fahr, E.: Neuro-Fuzzy Recognition of KComplexes in Sleep EEG Signals. *IEEE*, 789–790 (1997)
25. Akin, A., Akgul, T.: Detection of Sleep Spindles by Discrete Wavelet Transform. *IEEE*, 15–17 (1998)

Part V

**Special Session Soft Computing in
Computer Graphics, Imaging, and Vision**

Prostate Boundary Detection in Ultrasound Images Based on Type-II Fuzzy Sets and Modified Fuzzy C-Means

About Ella Hassanien, Gerald Schaefer, and Hameed AlQaheri

Abstract. Estimation of prostate location and volume is essential in determining a dose plan for ultrasound-guided brachytherapy, a common prostate cancer treatment. However, manual segmentation of the prostate area is difficult, time consuming and prone to variability. In this paper, we present an automated approach to prostate boundary detection in ultrasound images. First, we utilise type-II fuzzy sets to enhance the contrast of the prostate image. This is followed by a modified fuzzy c-means clustering algorithm in order to detect the actual boundary of the prostate pattern. To evaluate the performance of our approach, we present tests on different prostate ultrasound images and show that the overall accuracy offered by the employed approach is high.

1 Introduction

Prostate cancer is one of the most frequent cancers in the male population, and is a major cause of mortality in developed countries. Detection of prostate carcinoma at an early stage has been shown to be crucial for successful treatment (Scheipers et al, 2001). Different types of diagnostics, such as digital rectal examination (DRE) and prostate-specific antigen (PSA), are used to for such early detection. However, DRE may miss small tumors, while PSA values are dependent on several factors that are caused not only by prostate cancer (Han et al, 2008; Martinez et al, 2004). Therefore, diagnosis often

About Ella Hassanien

Information Technology Department, Cairo University, Egypt

Gerald Schaefer

Department of Computer Science, Loughborough University, U.K.

Hameed AlQaheri

Department of Quantitative Methods and Information Systems, Kuwait University, Kuwait

also relies on imaging methods for accurate localisation and staging of the disease (Martinez et al, 2004; Perez-Cortes et al, 2002).

Several imaging types can be used for such an approach. The use of transrectal ultrasound (TRUS) imaging has become common for this purpose due to its ability to visualise the prostate gland with no injurious effects, its low cost and its real time characteristic. While TRUS is currently the most widely used imaging method, it has a relatively low predictive value in detecting cancer due to the considerable overlap of benign and malignant lesion characteristics as well as the fact that its predictive ability is highly dependent on the radiologist's interpretation. To improve the performance, a computer-aided diagnosis system, which should provide useful information in real time to the radiologist, is required in order to allow for more accurate and efficient diagnosis with lower error rates. Unfortunately, no TRUS computer-aided diagnosis system for prostate cancer detection exists until now (Han et al, 2008; Martinez et al, 2004; Houston et al, 1995).

Accurate detection of the prostate boundary in ultrasound images is crucial for clinical applications, such as exact placement of needles during biopsy, accurate prostate volume measurement from multiple frames, construction of anatomical models used in treatment planning, and estimation of tumor borders. The underlying images are the result of reflection, refraction and deflection of ultrasound beams from different types of tissues with different acoustic impedance (Hirano et al, 2002). Unfortunately, ultrasound images suffer from low contrast which makes it difficult to determine the boundaries between the prostate and background. Also, speckle and weak edges make the ultrasound images inherently difficult to segment. Furthermore, the quality of the image depends on the type and particular settings of the imaging device.

In this paper, we present an automated algorithm to prostate boundary detection in TRUS images. A type-II fuzzy set approach for image enhancement is proposed. This is followed by a modified fuzzy c-means clustering algorithm in order to delineate the boundary of the prostate pattern.

The rest of the paper is organised as follows. Section 2 gives a brief introduction to the fuzzy image processing and type-II fuzzy sets. Section 3 presents our proposed prostate ultrasound image enhancement and boundary detection approach in detail. Experimental results are discussed in Section 4 while Section 5 concludes the paper.

2 Background

Various successful methods to analyse medical images are based on computational intelligence techniques (Schaefer et al, 2009). In our work we also follow this approach and employ fuzzy image processing concepts to extract useful information from TRUS images.

2.1 Fuzzy Image Processing

Fuzzy image processing gathers all approaches that understand, represent and process the images, their segments and features using fuzzy sets.

An image I of size $M \times N$ pixels having gray levels g ranging from 0 to $L - 1$ can be considered as an array of fuzzy singletons, each having a value of membership denoting its degree of brightness relative to some levels. Using fuzzy set notation, we can represent the image I as

$$I = \bigcup_{ij} \frac{\mu(g_{ij})}{g_{ij}}, i = 0, 1, \dots, M - 1, j = 0, 1, \dots, N - 1 \quad (1)$$

where g_{ij} is the intensity of the (i, j) -th pixel and μ_{ij} is its membership value. The membership function characterises a suitable property of image (e.g. edginess, darkness, textural properties) and can be defined globally for the whole image or locally.

2.2 Type-II Fuzzy Sets

Fuzzy logic systems usually employ type-I fuzzy sets and represent uncertainty by numbers in the range $[0,1]$ which are referred to as degrees of membership. Type-II fuzzy sets are an extension of type-I fuzzy sets with an additional dimension that represents the uncertainty about the degrees of membership. Type-II fuzzy sets are therefore useful in circumstances where it is difficult to determine the exact membership function for a fuzzy set. Type-I membership functions are precise in the sense that once they have been chosen all the uncertainty disappears. In contrast, type-II membership functions are fuzzy themselves. The simplest form of type-II fuzzy sets are interval type-II sets whose degree of membership are intervals with secondary membership degree of 1.0 (Ensafi and Tizhoosh, 2005).

Type-II fuzzy logic systems work as follows. Crisp inputs are first fuzzified into either type-0 (known as singleton fuzzifications) or type-I fuzzy sets, which then activate the inference engine and the rule base to produce output type-II fuzzy sets. These type-II fuzzy sets are then processed by a type-reducer which combines the output sets and then performs a centroid calculation, leading to an interval type-II fuzzy set called the type-reduced set. A defuzzifier then defuzzifies the type-reduced set to produce crisp outputs (Ensafi and Tizhoosh, 2005).

3 Prostate Boundary Detection Scheme

Ultrasound imaging techniques are widely used in medical diagnosis (Martinez et al, 2004; Han et al, 2008). Its noninvasive nature, low cost, portability, and real-time image formation make it an attractive means for medical

diagnosis. However, one of the limitations is rather poor image quality affected by speckle noise. Ultrasound images are therefore difficult to diagnose because of the existence of speckle which hampers the perception and the extraction of fine details from the image (Saad, 2007), while speckle reduction remains a difficult problem due to the lack of reliable models to estimate noise. To increase the efficiency of the classification and prediction process, a pre-processing stage should be considered to enhance the quality of the input prostate images before feature extraction and classification. In this paper, we present such an algorithm based on type-II fuzzy sets to enhance the contrast of TRUS prostate images. A modified fuzzy c-means segmentation algorithm is then used to perform region of interest (ROI) identification and to detect the boundary of the prostate pattern.

3.1 Type-II Fuzzy Set Contrast Enhancement

Type-II fuzzy sets are obtained by blurring a type-I membership function, using interval-based sets to construct the type-II set by defining the lower and upper membership values

$$\mu_{\text{LOWER}}(x) = \mu(x)^2 \quad (2)$$

and

$$\mu_{\text{UPPER}}(x) = \mu(x)^{0.5} \quad (3)$$

where $0 \leq \mu(x) \leq 1$ is the membership function for x .

Algorithm 1 shows the main steps of the type-II fuzzy set image enhancement technique we employ.

3.2 Modified Fuzzy C-Means Clustering

Fuzzy c-means (Zhou et al, 2009) is based on the idea of finding cluster centers by iteratively adjusting their positions and evaluation of an objective function which is typically defined as

$$E = \sum_{j=1}^C \sum_{i=1}^N \mu_{ij}^k \|x_i - c_j\|^2 \quad (7)$$

where μ_{ij}^k is the fuzzy membership of sample (or pixel) x_i and the cluster identified by its centre c_j , and k is a constant that defines the fuzziness of the resulting partitions.

E can reach the global minimum when pixels nearby the centroid of corresponding clusters are assigned higher membership values, while lower membership values are assigned to pixels far from the centroid. The membership is proportional to the probability that a pixel belongs to a specific cluster

Algorithm 1. Type-II fuzzy set image enhancement

Input: TRUS prostate image**Output:** Type-I and Type-II fuzzy enhanced images of the input image**Processing:****Step 1: Fuzzy hyperbolisation****for** each grey level **do**

Compute fuzzy type-I membership value

$$\mu(g_{mn}) = \frac{g_{mn} - g_{min}}{g_{max} - g_{min}} \quad (4)$$

 where g_{min} and g_{max} are the image minimum and maximum grey level.**end for****Step 2: Type-I fuzzy enhanced image calculation**

Compute new grey levels of the enhanced image using

$$\tilde{g}_{mn} = \frac{L-1}{e^{-1}-1} \times [e^{-\mu(g_{mn})^\beta} - 1] \quad (5)$$

 where the parameter β as a fuzzifier is set to 1.7 and the number of grey levels L is set to 256.**Step 3: Calculation of type-II fuzzy membership function** Compute $\mu_{\text{LOWER}}(x)$ and $\mu_{\text{UPPER}}(x)$ using Equations (2) and (3).

Divide the image into sub-images.

for every grey level value **do** Calculate a window of size 21×21 .

Compute type-II fuzzy membership function using

$$\mu_{TII}(g_{mn}) = (\mu_{\text{LOWER}} \times \alpha) + (\mu_{\text{UPPER}} \times (\alpha - 1)) \quad (6)$$

 where $\alpha = \frac{g_{mean}}{L}$.**end for****Step 4: Fuzzy type-2 enhanced image calculation** Compute the new grey levels of the enhanced image using Equation (5).

where the probability is only dependent on the distance between the image pixel and each independent cluster centre. The membership functions and the cluster centres are updated by

$$\mu_{ij} = \frac{1}{\sum_{m=1}^C \left(\frac{\|x_i - c_j\|}{\|x_j - c_m\|} \right)^{2/(k-1)}} \quad (8)$$

and

$$c_j = \frac{\sum_{i=1}^N \mu_{ij}^k x_i}{\sum_{i=1}^N \mu_{ij}^k} \quad (9)$$

Algorithm 2. Modified fuzzy c-means clustering**Input:** Enhanced prostate image, set of initial values of centroids**Output:** Set of clustered regions**Processing:**

repeat

 for $i=1$ to C do

Update the partition matrix (membership calculation).

$$D_{ik} = \|x_k - \lambda_i * c_k\|^2.$$

$$\phi_i = \sum_{y_r \in N_k} \|y_r - \lambda_r * v_i\|^2.$$

$$v_{ik}^* = \frac{1}{\sum_{j=1}^c \left(\frac{D_{ik} + \frac{\alpha}{N_R} \phi_i}{D_{jk} + \frac{\alpha}{N_R} \phi_j} \right)^{\frac{1}{p-1}}}$$

 where N_R stands for the cardinality of the set of neighbours that exist in a window around x_k . Influence of N_R is controlled by the parameter α .

end for

Update the centroid.

$$\text{Estimate } v_i^* = \frac{\sum_{k=1}^N u_{ik}^p \left[(y_k - \lambda_k) + \frac{\alpha}{N_R} \sum_{y_r \in N_k} (y_r - \lambda_r) \right]}{(1+\alpha) \sum_{k=1}^N u_{ik}^p}.$$

$$\text{Estimate a new gain filed } \lambda_j = \frac{\sum_{i=1}^c u_{ik}^p y_j v_i}{\sum_{i=1}^c u_{ik}^p}.$$

until $\|V_{new} - V_{old}\| < \epsilon$ where ϵ is the threshold set by the user and V is a vector of cluster centers.

The fuzzy c-means objective function is minimised when high membership values are assigned to pixels whose intensities are close to the centroid of its particular class, and low membership values are assigned when the pixel data is far from the centroid. The algorithm is computationally expensive and sensitive to noise. To address these problems, we use a modified fuzzy c-means clustering algorithm (Hassanien, 2007; Ahmed et al, 2003; Birgani et al, 2008) which is detailed in Algorithm 2.

4 Experimental Results

Our dataset consists of 212 TRUS images (98 cancer and 114 non-cancer cases). Due to the differences in conditions under which acquisition occurred, the images are not at the same level of contrast and we therefore apply the pre-processing step introduced in Section 3.1. Figure 1 shows examples of the results of the type-II fuzzy set-based image enhancement on four TRUS images. The first column of each row depicts the original prostate image, the second column shows the result of a locally adaptive type-I fuzzy approach presented in (Ensafi and Tizhoosh, 2005), while the third column gives the result of the new type-II fuzzy set approach. It is apparent from Figure 1 that the type-II algorithm clearly outperforms the type-I-based approach.

Figure 2 shows the segmented boundary result using obtained from the modified fuzzy c-means algorithm presented in Section 3.2. Figure 2 (a) and (b) show the original and the type-II fuzzy enhanced image respectively, while

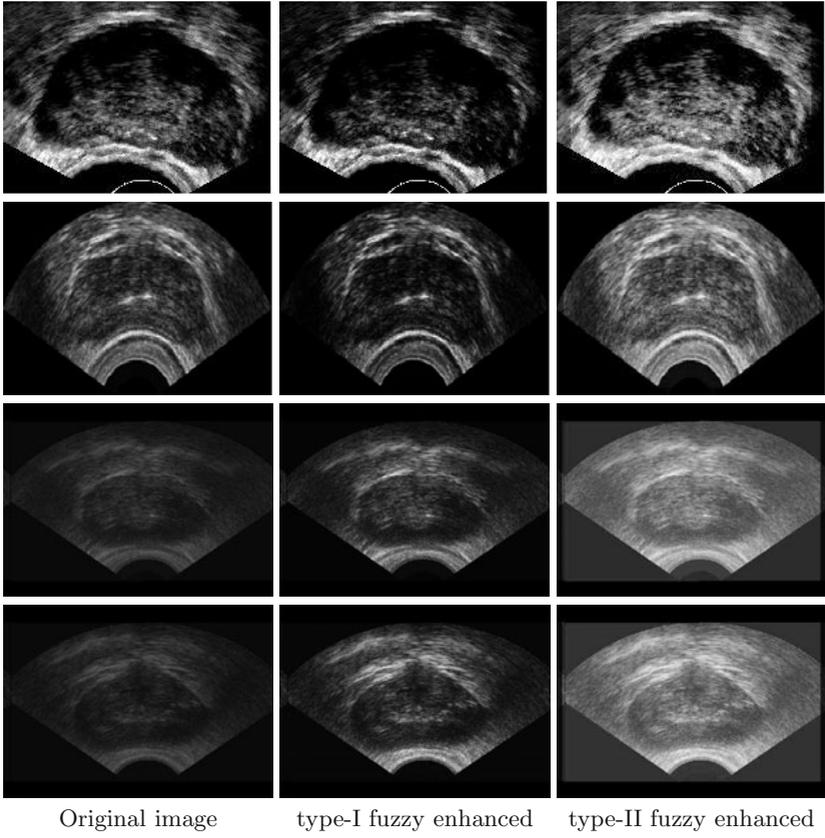


Fig. 1 Image enhancement results based on type-I and type-II fuzzy sets

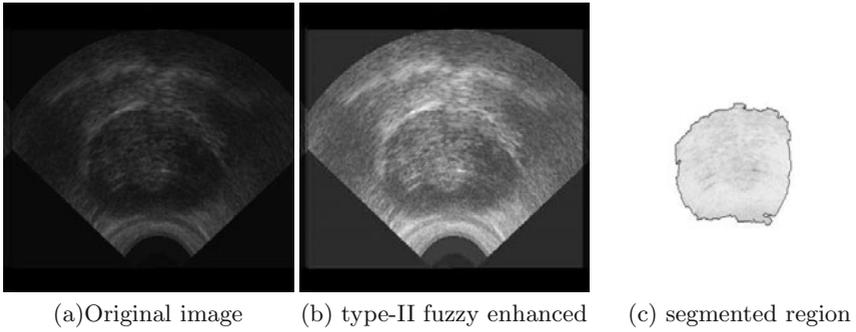


Fig. 2 Segmentation results

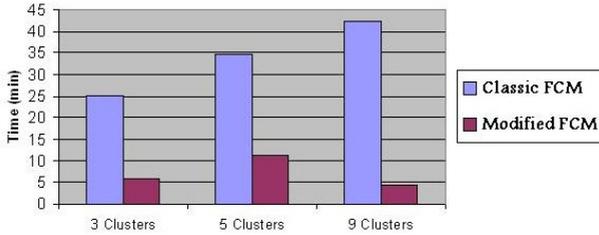


Fig. 3 Run-time comparison of conventional fuzzy c-means and our modified algorithm

(c) shows the resulting segmented boundary result. The average segmentation accuracy of our approach, established through measuring the similarity between the obtained results and a gold standard based on manual segmentation, is 2.14%, which confirms that it is sufficiently robust.

In Figure 3 we show the computation time of our modified c-means clustering algorithm compared with the classical fuzzy c-means. As can be observed, our approach is significantly faster than conventional fuzzy c-means.

5 Conclusions

In this paper we have presented an algorithm based on type-II fuzzy sets to enhance the contrast of TRUS prostate images. This is followed by a modified fuzzy c-means clustering algorithm in order to detect the boundary of the prostate pattern. To evaluate the performance of our approach, we presented tests on different prostate ultrasound images which show that the overall accuracy offered by the employed approach is high. Furthermore, the modified clustering algorithm was shown to be computationally more efficient compared to the conventional fuzzy c-means algorithm.

References

- Ahmed, M., Yamany, S., Nevin, M., Farag, A.: A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data. *IEEE Transactions on Medical Imaging* 21(3), 193–199 (2003)
- Birgani, P., Ashtiyani, M., Asadi, S.: MRI segmentation using fuzzy c-means clustering algorithm basis neural network. In: 3rd Int. Conference on Information and Communication Technologies: From Theory to Applications, pp. 1–5 (2008)
- Ensafi, P., Tizhoosh, H.: Type-2 fuzzy image enhancement. In: Kamel, M.S., Campilho, A.C. (eds.) *ICIAR 2005*. LNCS, vol. 3656, pp. 159–166. Springer, Heidelberg (2005)
- Han, S., Lee, H., Choi, J.: Computer-aided prostate cancer detection using texture features and clinical features in ultrasound image. *Journal of Digital Imaging* 21(suppl. 1), 121–133 (2008)

- Hassanien, A.: Fuzzy-rough hybrid scheme for breast cancer detection. *Image and Computer Vision Journal* 25(2), 172–183 (2007)
- Hirano, S., Sun, X., Tsumoto, S.: Dealing with multiple types of expert knowledge in medical image segmentation: A rough set style approach. In: *IEEE Int. Conf. on Fuzzy Systems*, vol. 2, pp. 884–889 (2002)
- Houston, A., Permkuma, S., Pitts, D.: Prostate ultrasound image analysis: Localization of cancer lesions to assist biopsy. In: *IEEE 8th Symposium on Computer Based Medical Systems*, pp. 94–101 (1995)
- Martinez, C., Oglio, M.D., et al.: Predictive value of PSA velocity over early clinical and pathological parameters in patients with localized prostate cancer who undergo radical retropubic prostatectomy. *Int. Braz. J. Urol.* 30(1), 12–17 (2004)
- Perez-Cortes, J., Juan, A., Vallada, E.: Textural analysis of prostate cancer in transrectal ultrasound images. In: *Biosignal 2002* (2002)
- Saad, A.: Visual enhancement of digital ultrasound images: wavelet versus Gauss-Laplace contrast pyramid. *Int. J. CARS* 2, 117–125 (2007)
- Schaefer, G., Hassanien, A., Jiang, J.: *Computational Intelligence in Medical Imaging: Techniques and Applications*. CRC Press, Boca Raton (2009)
- Scheipers, U., Pesavento, A., Ermert, H., J-Sommerfeld, H., Garcia-Schurmann, M., Kuhne, K., Senge, T., Philippau, S.: Ultrasound multifeature tissue characterization for the early detection of prostate cancer. In: *IEEE Ultrasonics Symposium*, pp. 1265–1268 (2001)
- Zhou, H., Schaefer, G., Shi, C.: Fuzzy c-means techniques for medical image segmentation. In: *Fuzzy Systems in Bioinformatics and Computational Biology. Studies in Fuzziness and Soft Computing*, pp. 257–271. Springer, Heidelberg (2009)

Distributed Multimedia Computing Using Intelligent Agents

Gerald Schaefer and Roger Tait

Abstract. Multimedia data is ever increasing, and efficient and effective solutions in multimedia computing and processing are therefore highly sought after. In this paper, we address the problem of analysing and processing multimedia data in a distributed fashion using multiple intelligent agents that communicate via a blackboard interface. We propose a system with three different kinds of agents. A Distributor agent splits multimedia data into smaller segments before placing them on the blackboard. Worker agents retrieve these segments and process them in a distributed fashion. An Accumulator agent then reconstructs the processed multimedia output. Co-ordination of agents is achieved by means of reactive behaviour and communication via the blackboard, thus removing the need for a dedicated control module and associated overheads.

1 Introduction

Multimedia, typically defined as the integration of multiple forms of media, plays an important role in many areas. The currently accepted understanding of multimedia computing involves a variety of media, such as still images, videos, sound, music and text, using a computer as storage device, delivery controller, delivery medium and interaction platform. Current research of multimedia processing is shifting from coding (MPEG-1,2,4) towards automatic recognition and annotation (MPEG-7) focussing on techniques for object-based representation and coding, segmentation and tracking, pattern detection and recognition, multimodal signal fusion, conversion and synchronisation, as well as content-based indexing and subject-based retrieval and browsing.

Gerald Schaefer

Department of Computer Science, Loughborough University, U.K.

Roger Tait

University of Cambridge, U.K.

Due to the large amounts of data involved and the related high computational requirements for processing and analysing these data, efficient and effective solutions for multimedia computing are highly sought after. One approach to increased efficiency is the application of parallel processing paradigms to the analysis and processing of multimedia data. In this paper we propose such an approach based on a blackboard communication strategy and the use of multiple intelligent agents for efficient and effective multimedia computing.

2 Parallel Multimedia Processing

Parallel processing approaches to multimedia processing can in general be divided into two classes: tightly-coupled and loosely-coupled methods. A tightly-coupled architecture comprises a machine with multiple processors. Data distribution is not required as all processors share the same main memory and can therefore work on the same data concurrently. As a consequence, this approach largely eliminates the need for explicit message passing between concurrent tasks. Multi-threaded programming allows applications to branch into independent and potentially concurrent threads and is not restricted to shared-memory multiprocessor architectures. In general, multi-threaded applications are well suited to multiprocessor architectures since individual threads can run concurrently on all available processors. As multi-threaded applications often share the same address space, they cause considerably fewer overheads than the creation of an equivalent number of processes. Although a minimal cost is associated with the creation and handling of multiple threads, the performance gain sought must outweigh all overheads in order for the implementation to be useful (Lewis and Berg, 1996).

In contrast, in a loosely-coupled architecture the parallel processing of multimedia data consists of four main steps: data distribution, local processing, data transfer during processing, and segment accumulation. Distribution is the process of dividing the data segments each of which is then assigned to a unique processor (Taniguchi et al., 1997). Under a duplicate distribution scheme each processor is sent an exact copy of the original data although more complex schemes can also be adopted (Nicolescu and Jonkel, 2000). Once distributed, each processor applies local processing to its allocated segments. Finally, after application of the parallel algorithm, segments are accumulated to form the processed data.

Inter-processor communication is required when data allocated to other processors are needed and can be categorised into groups based on their pattern of data access (Seinstra et al., 2002). These patterns also represent a strategy for synchronisation between communicating processors. One-to-one access is common in such functionality as subtraction and multiplication, e.g. when an output pixel of a video stream maps directly to a pixel in the input frame. Alternatively, a one-to-many relationship exists e.g. in neighbourhood

operators, such as image edge detection filters, or in situation where multiple video frames are required for processing. Naturally, the handling and transmission of non-contiguous data differs from data stored as contiguous blocks. Data stored randomly in memory causes additional overheads due to its packing into a contiguous buffer before transmission (Hoare, 1985).

3 Distributed Multiagent Blackboard Architecture

Blackboard systems are based on the analogy of a group of experts working together to solve a common problem by writing their ideas onto a common blackboard (Engelmore and Morgan, 1988). Such architectures have emerged from their 1970s origins as a suitable host for multiagent implementations (Jufeng et al., 2004). Traditional implementations are constructed from three distinct components including a blackboard, expert agent modules (also known as knowledge sources), and a control unit. The blackboard represents an area of shared memory where agents can store and retrieve information. Also known as the scheduler, a control unit monitors changes on the blackboard and is used to determine the focus of attention. The focus of attention can be described as the selection of an agent to be activated or selection of a solution to pursue. Distributed blackboard systems can be hosted on both tightly- and loosely-coupled architectures.

Suited to distributed implementation, multiagent systems offer the possibility of directly representing the individual components of a complex system since the agents behave autonomously and interact with other system components (Murch and Johnson, 1998). Specialist behaviour encapsulated within an agent gives it the ability to adapt, interact and evolve within the environment in which it exists. An agent makes decisions based upon memory, internal state and information received from other agents. If multiple agents are hosted on independent computers within the same network, communication allows for interaction and collaboration, thus permitting parallel processing capabilities. Synchronous communication requires an agent to suspend communications until all processing tasks are complete (Ferber, 1999). In contrast, agents that communicate and continue to process, are operating in an asynchronous environment. Globally, the propagation of messages to all agents is achieved by means of a broadcast mechanism.

4 Distributed Intelligent Agents for Multimedia Processing

In this paper, we propose the use of multiple intelligent agents distributed via a blackboard architecture for effective and efficient processing of multimedia data. As mentioned above, blackboard architectures provide a co-ordinated and distributed problem solving environment. In our approach we employ

DARBS (Distributed Algorithmic and Rule-based Blackboard System), a distributed blackboard architecture based on a client/server model, in which the server functions as the blackboard and client modules as agents (Nolle et al., 2001). DARBS was also used as the underlying system of our earlier work on parallel image processing (Tait et al., 2007), and in this paper we extend the concepts derived there to generic multimedia data and processing.

Our DARBS framework for distributed multimedia processing, comprises three types of DARBS agents:

- A Distributor agent which splits multimedia data into segments which are then placed on the blackboard.
- Worker agents which retrieve segments from the blackboard and perform local processing.
- An Accumulator agent which collects processed segments and constructs the processed data, while co-ordinating Worker agent activities.

In the following, we discuss the various technical aspects of our framework.

4.1 *Blackboard Partitioning*

Storing the required multimedia data on the blackboard ensures that all agents have equal access to it. Division of the blackboard into partitions that correspond to agent types simplifies management of agent activities. In addition, partitioning is also used to balance communication and processing workloads. To achieve this, the blackboard is divided into the following partitions:

- A *Distributor control partition* controls division of the data into segments.
- *Worker control partitions* are used for the processing of segments.
- Accumulation of segments and supervision of Worker agent activities is achieved by means of an *Accumulator control partition*.
- System variables are maintained in a *Parameters partition*.
- The *Multimedia data partition* holds the partitioned multimedia data.

4.2 *Blackboard Data and Communication*

Clearly, any type of multimedia data can be contained on the blackboard. Division into segments though depends on the type of data and is handled as follows:

- Image data is segmented into (possibly overlapping) image sub-blocks.
- Video data is segmented either into single frames or into collections of adjacent frames.
- Segmentation of audio data is time-based, i.e. the stream is segmented into fixed or variably timed intervals.

Other schemes can be easily added by modifying the behaviour of the Distributor and Accumulator agents, or adding additional Distributor/Accumulator agents.

To promote efficient processing of agent queries and hence increase overall performance, careful consideration was given to the information stored on and communicated via the blackboard. Components include, among others, an identification tag, parameters associated with a multimedia processing operator, and the number of segments.

The content of a *Worker control partition* is of particular interest. The information string [Initialised] is added to the partition by a Worker agent, once it has been initialised. [Start] is then added by the Accumulator agent as it co-ordinates the start of Worker agent activities. When an unprocessed segment has been retrieved from the blackboard, [Start] is replaced with [Fetched]. In turn, [Fetched] is replaced by [Processed] as soon as the multimedia segment has been processed by a Worker agent. In a final step, [Processed] is replaced with [Stop] and the processed segment is returned to the blackboard. By replacing previous information strings with current strings, the search of the *Worker control partition* is kept to a minimum.

The format of an information string is also of importance. In anticipation of more complex tasks, parameters contained within an information string are arranged in order of frequency of use, with the most regularly used parameters appearing first. This ordering of parameters allows for fast and efficient query of an information string by interested agents.

5 Agent Implementation

In this section we describe the three types of agents in more detail and explain how they interact to provide an effective and efficient framework for multimedia data analysis and processing.

5.1 Distributor Agent

The Distributor agent first clears all the data from the blackboard. A segment list which is locally maintained by the agent is also initialised in preparation for new data. Selection of a multimedia source is performed through an integrated file browser which also supports preview/playback of image, video and audio data. Alternatively, a batch scheduler is implemented which allows the selection of multiple multimedia data files to be processed using the framework. System data, in the form of a parameters information string is constructed and then added to the *Parameters partition*. The information string contains multimedia processing operator parameters, the number of segments, and other, datatype dependent information (e.g. for images the size of borders between segments to allow overlapping of image subblocks). Next, using the number of segments, the source data is divided into segments.

Resulting segments are then compressed, accumulated in the local segment list, and sent to the *Multimedia data partition* on the blackboard.

5.2 *Worker Agent*

Division of the source multimedia data into segments means that the tasks performed by each Worker agent are identical. As a consequence, the implementation of each Worker agent is identical and depends solely on the task to be performed.

In order to achieve co-ordination of multiple identical Worker agents, each Worker agent is numbered. Because Worker agents act for the majority of their time upon partitions with corresponding numbering, the numbering scheme is also used to assign data segments to Worker agents. As an example, Worker n agent adds and removes information strings from the *Worker n control partition*. Worker n agent also fetches segment n from the *Multimedia data partition* and returns the processed segment to the partition.

Retrieval of the parameter information string from the *Parameters partition* is the first task of a Worker agent. Once retrieved, parameters associated with the processing operator are extracted. The Worker n agent then waits for a start information string to appear in the *Worker n control partition*. This start information string acts as a trigger mechanism, which prevents retrieval of segment n before it has been placed in the *Multimedia data partition*. Once triggered, segment n is retrieved from the blackboard. To reduce the overheads caused by redundant data, whenever a segment is retrieved it is removed from the *Multimedia data partition*. The retrieved segment is then decompressed and processed using the process dependant operator parameters. Once processed, segment n is compressed and returned to the *Multimedia data partition*. In a final step, the start information string in the *Worker n control partition* is replaced with a stop string and the Worker n agent terminates.

5.3 *Accumulator Agent*

The initial task performed by the Accumulator agent is clearance of a locally maintained segment list, in preparation for new data. The parameter information string is also fetched from the *Parameters partition*. Once retrieved, the number of segments, and other data are extracted. A start information string is then propagated to all *Worker control partitions*. These cause the waiting Worker agents to commence processing. The Accumulator agent now waits for stop information strings to appear in all *Worker control partitions*. Once this has occurred, processed segments are retrieved from the *Multimedia data partition*. Each retrieved segment is decompressed and added to the locally maintained processed segment list. Once all processed segments have been accumulated, the resulting processed output data is constructed.

Whenever a stop information string is added to a *Worker control partition*, the Accumulator agent is forced to restart. Upon restarting, addition of start information strings to all *Worker control partitions* does not reoccur. Without this fire once mechanism, both Accumulator and Worker agents would endlessly restart as they repeatedly modify the *Worker control partitions*.

6 Conclusions

In this paper we have presented a framework for the distributed processing of multimedia data using multiple intelligent agents. Based on a loosely-coupled blackboard communication architecture, three types of intelligent agents are employed to allow for effective and efficient distributed multimedia computing. A Distributor agent splits the incoming multimedia data into smaller segments. These segments are then picked up by Worker agents which perform local processing and return the processed segments to the blackboard. This task is performed in parallel by multiple agents. When all individual segments are processed, an Accumulator agent assembles the processed multimedia data.

Due to its modular structure, our framework is suitable for any kind of multimedia data, both existing and to come. The only adjustments that need to be made when a new datatype is introduced are modifications to the Distributor and Accumulator agents to define how the data is to be split into segments. Similarly, adding an additional type of multimedia processing (e.g. a new video processing technique) only requires modification or adding of a new Worker agent that implements the desired behaviour. Consequently, our approach is versatile in use and easily adaptable to new requirements.

References

- Engelmore, R., Morgan, T.: Blackboard Systems. Addison-Wesley, Reading (1988)
- Ferber, J.: Multiagent Systems: An Introduction to Distributed Artificial Intelligence. Addison-Wesley, Reading (1999)
- Hoare, C.: Communicating Sequential Processes. Prentice-Hall, Englewood Cliffs (1985)
- Jufeng, W., Hancheng, X., Xiang, L., Jingping, Y.: Multiagent based distributed control system for an intelligent robot. In: IEEE International Conference on Services Computing, pp. 633–637 (2004)
- Lewis, B., Berg, D.: A guide to multi-threaded programming. Prentice-Hall, Englewood Cliffs (1996)
- Murch, R., Johnson, J.: Intelligent Software Agents. Prentice-Hall, Englewood Cliffs (1998)
- Nicolescu, C., Jonker, P.: Parallel low-level image processing on a distributed memory system. In: 15th Workshop on Parallel and Distributed Processing, pp. 26–233 (2000)

- Nolle, L., Wong, K., Hopgood, A.: DARBS: A distributed blackboard system. In: Research and Development in Intelligent Systems XVIII, pp. 161–170 (2001)
- Seinstra, F., Koelma, D., Geusebroek, J.: A software architecture for user transparent parallel image processing. *Parallel Computing* 328, 967–993 (2002)
- Tait, R., Schaefer, G., Hopgood, A.: iDARBS - a distributed blackboard architecture for parallel image processing. *Int. Journal of Information Technology and Intelligent Computing* 2(4) (2007)
- Taniguchi, R., Makiyama, Y., Tsuruta, N., Yonemoto, S., Arita, D.: Software platform for parallel image processing and computer vision. In: *Parallel and Distributed Methods for Image Processing. Proceedings of SPIE*, vol. 3166, pp. 2–10 (1997)

Object Recognition Using Simulated Evolution on Fourier Descriptors

Muhammad Sarfraz, Jawad Usman Arshed, Sajjad Ahmad Madani, and Muhammad Iqbal

Abstract. This work presents a study on object recognition of isolated objects. Fourier descriptors have been used as features of the outlines of the objects under discussion. The problem of selecting the best descriptors has been formulated as an optimization problem and solved using Evolution.

1 Introduction

Several techniques have been developed that derive features for object recognition and representation [1-10]. Fourier descriptors [1-3, 6-7], like Moment descriptors [8], have been frequently used as features for image processing, remote sensing, shape recognition and classification. Fourier Descriptors can provide characteristics of an object that uniquely represent its shape.

Fourier descriptors have useful properties [1-3]. They are invariant under similarity transformations like translation, scaling and rotation. The objects having these kinds of transformations can be easily recognized using some recognition algorithms with Fourier descriptors as invariant features.

This paper has used Fourier descriptors, with different combinations, for the recognition of objects captured by an imaging system which may transform, make noise or can have occlusion in the images. An extensive experimental study, similar to the moment invariants [6] and Fourier descriptors [6], has been made using similarity measures in the process of recognition. These measures include Euclidean Measure and Percentage of error. Comparative study of various cases has provided very interesting observations which may be quite useful for the researchers as well

Muhammad Sarfraz

Department of Information Science, Adailiyah Campus, Kuwait University,
P.O. Box 5969, Safat 13060, Kuwait
e-mail: prof.m.sarfraz@gmail.com

Jawad Usman Arshed · Sajjad Ahmad Madani · Muhammad Iqbal

Department of Computer Science, CIIT, COMSATS, Abbottabad, Pakistan
e-mail: {jawadusman, madani, iqbal}@ciit.net.pk

as practitioners working for imaging and computer vision problem solving. Although the whole study has been made for bitmap images, but it can be easily extended to gray level images.

In this study, the problem of selecting the best descriptors, has been formulated as an optimization problem. Simulated Evolution (SE) technique has been mapped and used successfully to have an object recognition system using minimal number of Fourier descriptors. The goal of the proposed optimization technique is to select the most helpful descriptors that will maximize the recognition rate. The proposed method will assign, for each of these descriptors, a weighting factor that reflects the relative importance of that descriptor.

The outline of the remainder of the paper is as follows. Getting of bitmap images and their outline is discussed in Sections 2. Section 3 deals with the study of Fourier descriptors together with the concepts of similarity measures explained. Results and analysis of simple object recognition system are given in Section 4. Proposed study for object recognition problem has been devised in Section 5. This section also contains detailed experimental study and analyses together with interesting observations during the experimental study. Finally, Section 6 concludes the paper.

2 Getting Bitmap Image and Boundary

Bitmap image of a character can be obtained by creating a bitmap character on some program like *Paint* or Adobe Photoshop. Alternatively an image drawn on paper can scan and store it as bitmap. We used both methods. The quality of bitmap image obtained directly from electronic device depends on the resolution of device, type of image (e.g. bmp, jpeg, tiff etc), number of bits selected to store the image etc. The quality of scanned image depends on factors such as quality of image on paper, scanner and attributes set during scanning. Figure 1(a) shows the bitmap image of a character.



Fig. 1 (a) Bitmap image, (b) Outline of the image

In order to find boundary of bitmap image, first its chain code is extracted [10]. Chain codes are a notation for recording the list of edge points along a contour. The chain code specifies the direction of a contour at each edge in the edge. From chain coded curve, boundary of the image is found [16]. The algorithm returns number of pieces in the image. Figure 1(b) shows detected boundary of the image of Figure 1(a).

3 Fourier Theory and Similarity Measure

To characterize objects we use features that remain invariant to translation, rotation and small modification of the object's aspect. The invariant Fourier descriptors [1-3] of the boundary [10-15] of the object can be used to identify an input shape, independent on the position or size of the shape in the image.

In the scope of this research, the Fourier transform technique is used for shape description in the form of Fourier descriptors. The shape descriptors generated from the Fourier coefficients numerically describe shapes and are normalized to make them independent of translation, scale and rotation.. The procedure has been implemented in such a way that the boundary of the image is treated as lying in the complex plane. So the row and column co-ordinates of each point on the boundary can be expressed as a complex number. For details, the reader is referred to [5, 14].

This paper implements a simple classifier that calculates similarity measure, namely Euclidean Distance (ED), of the corresponding Fourier descriptors of the input shape. These shapes are contained in the database in the form as shown in Figure 1. Thus, if two shapes, A and B , produce a set of values represented by $a(i)$ and $b(i)$ then the distance between them can be given as $c(i) = a(i) - b(i)$. If $a(i)$ and $b(i)$ are identical then $c(i)$ will be zero. If they are different then the magnitudes of the components in $c(i)$ will give a reasonable measure of the difference. The easiest way is to treat $c(i)$ as a vector in a multi-dimensional space, in which case its length, which represents the distance between the objects, is given by the square root of the sum of the squares of the elements of $c(i)$, i.e

$$c(i) = \sqrt{\sum_{i=1}^n (a(i) - b(i))^2} . \quad (1)$$

In this study, n is the number of FDs considered, $a(i)$ is the i th FD of the template image, and $b(i)$ is the i th FD of the test image. A tolerable threshold ρ is selected to decide a test object recognized. This threshold is checked against the least value of the selected similarity measure.

4 Results and Analysis

The simple recognition system, see Figure 2 without the oval surrounded part, is tested by generating the test objects by translating, rotating, scaling, adding noise, and adding occlusion to the model objects contained in a database of different sizes. The test objects were randomly rotated, translated, and scaled. Some were considered without scale of their model sizes. About 60 test objects were used for each of the experiments for testing similarity transformation. The salt & pepper noise [10] of different densities is added to the objects for generating the noisy test objects. Median filter [10] was used in the experiments to filter the noise, so that the noise remains on the boundary of the object.

Table 1 Recognition rates of Fourier descriptors using Euclidean distance

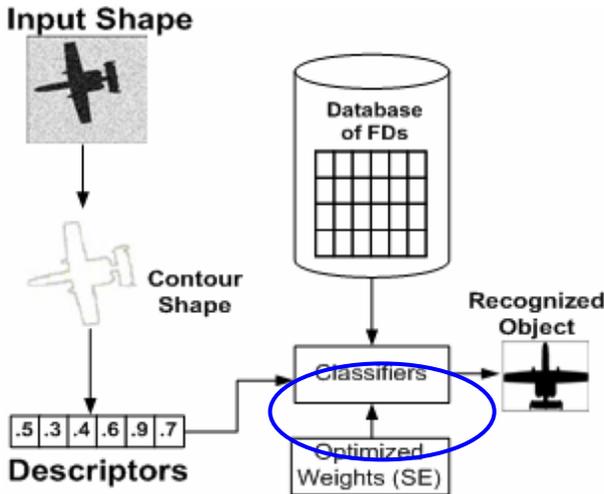
Number of FDs	4	6	11	18	22	29
Transformations	71.67%	83.33%	93.33%	90%	93.33%	95%
Noise	75%	93.75%	93.75%	93.75%	93.75%	93.75%
Occlusion	5%	8.33%	20%	18.33%	23.33%	23.33%

As can be seen in the experiments (Table 1), FDs are not promising for the recognition of occluded objects. Around 20% occlusion was added into the objects to make tests. One can see, in Table 1, the FDs range from 4 to 29. With the increase of number of FDs the recognition has increased in general.

5 Proposed Technique Using Simulated Evolution (SE)

For the optimization, we have used Simulated Evolution optimization technique. Simulated evolution (SE) is a general optimization algorithm based on the analogy between optimization and evolutionary processes. Most of the applications of SE by other researchers were also to solve hard design automation problems. It has been successfully applied to such problems as standard cell placement, switch box routing, and circuit partitioning. There are very few applications in other areas of engineering.

As we may have thousands of descriptors for an image, the problem is to select the best descriptors which can improve the recognition rate. The problem of selecting the best descriptor can be formulated as an optimization problem. The pictorial representation of the proposed technique is shown in Figure 2 (with the

**Fig. 2** Pictorial Description of the method

oval surrounded part). It is clear from the figure that the optimized weights are used with FDs in the process of recognition. These weights obtained from SE show the relative importance of these descriptors.

5.1 Optimization of the Feature Vector Using Simulated Evolution

The problem of selecting the best descriptors can be formulated as an optimization problem. The goal of the optimization is to select the most helpful descriptors that will maximize the recognition rate and assign for each of these descriptors a weighting factor that reflects the relative importance of that descriptor.

As we need to select the best descriptors, this turns the problem into an optimization problem. Thus, we need to define an objective function with respect to the simulated evolution. The objective function in this case, is made up of the following two facts:

- the recognition rate,
- the number of useful descriptors.

In other words, it is required to maximize the recognition rate using the minimum number of descriptors. The Simulated Evolution (SE) requires the following four steps:

Step 1: (Initialization) The first step for simulated evolution in the optimization process is initialization. In this step various parameters are initialized to their desired value. In our simulation we have set the following parameters.

- Bias is set between 0 and 0.2.
- Number of iterations are 40 but we also have set the stopping criteria. If our stopping criteria meets during the specified iterations the simulation ends. Otherwise it goes for the number of iterations specified.
- Numbers of trials are used in between 10 to 20.
- We have also set the stopping criteria which depends on the Hits used in the simulation.

Step 2: (Evaluation) The second step of simulated evolution is to evaluate the goodness of each particle as in our case we have the weights used as particles, so we compute goodness against each of the particle. Goodness against each particle shows the relative importance of those weights, and is given by:

$$g = 1 - \min(PE), \quad (2)$$

where g is the goodness and PE is the percentage of errors of all the training images for a given set of weights.

Step 3: (Selection) The third and most important step of Simulated Evolution is selection. Selection takes as input the population “ P ” and estimated goodness of each individual g_i and divides the population P in to two sets, a selection set P_s

and a set Pr of remaining elements. The decision whether the individual i is assigned to the set Ps or Pr is based on their goodness g_i . Selection operator uses a selection function which takes g_i and B (selection bias) as input and return true or false which decide whether individual i belongs to Ps or Pr . The logical expression we have used in our selection function are as follows:

$$Random \leq 1 - g_m + B, \quad (3)$$

where $Random$ is the percentage of errors of all the training images for a given set of weights and B is the selection bias. In our simulations we have the best results when B is in the range from 0 to 0.2. On the basis of this logical expression the selection of individuals for the set Ps and Pr are decided.

Step 4: (Allocation) Allocation is the last but most important step of Simulated Evolution which has great impact on the quality of solution. It takes two sets Ps and Pr as input and generates a new population by applying mutation on each individual of Ps . Allocation alters all of the elements in the selected set Ps one after the other. The order as well as the type of mutation is problem specific. We have used the mutation up to 11 binary numbers and used the same order as we get individuals in Ps . After each trial Ps and Pr are updated and the every individual is set to its best location. These steps continue until the defined stopping criteria are met.

In this way, Simulated Evolution searches for the weights that result in the highest recognition rate. The goodness for each individual tries to search the best weights for each individual and check of hit rate makes the Simulated Evolution reach to the highest recognition with the minimum number of descriptors.

The SE algorithm, in this research, starts from an initial assignment. Then following an evolutionary based approach, it seeks to reach better assignments from one generation to the next. Initially, a population is created at random. The algorithm has three basic steps inside one main loop. These steps are Evaluation, Selection and Allocation as mentioned in the previous paragraphs. The three steps are executed in sequence until no noticeable improvement to the population goodness is observed after a number of iterations. For more detail, about general SE algorithm and detail about SE steps and operators used in SE, the reader is referred to [13].

5.2 Test Results Using Simulated Evolution

The proposed SE-based approach was implemented using MATLAB 7.1. In our implementation we have set bias in between 0 and 0.2, number of iterations are set to 40. In each iteration, we have used 10 to 20 trials. Also note that we have set the stopping criterion which is also checked during these iterations. The search process may also stop if stopping criteria is met during the specified number of iterations.

Table 2 Optimized weights for different numbers of Fourier descriptors

Experiment No.	1	2	3	4	5
Training set*	X	X	X	O	X, O, N
No. of FDs	7	11	6	6	11
Optimized Weights obtained	0.105	0.1193	0.1	0.1	0.169
	0.23	0.158	0.33	0.0929	0.3351
	0.0579	0.1478	0.1663	0.23	0.1923
	0.4057	0.7134	0.5208	0.58	0.7157
	0.9169	0.5221	0.7181	0.9275	0.2507
	0.4103	0.5962	0.5692	0.3911	0.9339
	0.8936	0.8162			0.1372
		0.9771			0.5216
		0.9329			0.9424
		0.4496			0.4374
	0.1722			0.4712	

Several experiments have been attempted to use SE to search optimum descriptor weights and then use different similarity measures as described above to check the recognition rate. These experiments are summarized in Tables 2 and 3.

Table 3 Total number of optimized weights used for different numbers of Fourier descriptors and the recognition rate using Euclidean distance

Experiment No.	1	2	3	4	5	
Training set*	X	X	X	O	X, O, N	
No. of FDs Used	7	11	6	6	10	
Recognition Rate	X	96.67	100	96.67	93.33	96.67
	N	93.75	93.75	93.75	100	93.75
	O	21	31	20	25	21

*X = transformed objects, O = occluded objects, N = noisy objects.

Table 2 demonstrates the computed optimized weights for different numbers of Fourier descriptors. These optimized weights are then used to check the recognition rate shown in Table 6. When a database of 60 transformed objects, is considered one can see that a much better result in Table 3 by using 7 descriptors as compared with Table 1 by using 29 descriptors. Similarly when we use 11 weighted descriptors it gives us 100 percent result which we did not achieve from un-weighted descriptors. Similar results can be seen for different experiments in the Table 3.

6 Conclusion and Future Work

This work has been reported to make a practical study of the Fourier descriptors to the application of Object Recognition. The ultimate results have variations depending upon the selection of number of FDs, noise, occlusion. The images used are all bitmapped images, further investigations are being done with some more complex images. The images that have to be recognized but failed to be recognized by most of the FD combinations have been analyzed using the methodology of Simulated Evolution. Use of Simulated Evolution, to find the most suitable descriptors and to assign weights for these descriptors, improved dramatically the recognition rate using the least number of descriptors.

References

1. Granlund, G.H.: Fourier Preprocessing for hand print character recognition. *IEEE Trans. Computers C-21*, 195–201 (1972)
2. Bertrand, O., Queval, R., Maître, H.: Shape Interpolation by Fourier Descriptors with Application to Animation Graphics. *Signal Processing* 4, 53–58 (1981)
3. Zahn, C.T., Roshkies, R.Z.: Fourier descriptors for plane closed curves. *IEEE trans. Compu.* 21, 269–281 (1972)
4. Bernier, T., Landry, J.-A.: A new method for representing and matching shapes of natural objects. *Pattern Recognition* 36, 1711–1723 (2003)
5. Zhang, J., Zhang, X., Krim, H., Walter, G.G.: Object representation and recognition in shape spaces. *Pattern Recognition* 36(5), 1143–1154 (2003)
6. Sarfraz, M., Al-Awami, A.T.A.: Object Recognition using Particle Swarm Optimization on Fourier Descriptors. In: Saad, A., Avineri, E., Dahal, K., Sarfraz, M., Roy, R. (eds.) *Soft Computing in Industrial Applications: Recent and Emerging Methods and Techniques*. *Advances in Soft Computing*, vol. 39, pp. 19–29. Springer, Heidelberg (2007)
7. Sarfraz, M.: Object Recognition using Fourier Descriptors: Some Experiments and Observations. In: Banissi, E., Sarfraz, M., Huang, M.L., Wu, Q. (eds.) *Computer Graphics, Imaging and Visualization – Techniques and Applications*, pp. 281–286. IEEE Computer Society, USA (2006), ISBN: 0-7695-2606-3
8. Sarfraz, M.: Object Recognition using Moments: Some Experiments and Observations. In: Sarfraz, M., Banissi, E. (eds.) *Geometric Modeling and Imaging – New Advances*, pp. 189–194. IEEE Computer Society, USA (2006)
9. Gorman, J.W., Mitchell, O.R., Kuhl, F.P.: Partial shape recognition using dynamic programming. *IEEE Transactions on pattern analysis and machine intelligence* 10(2) (March 1988)
10. Gonzalez, R., Woods, R., Eddins, S.: *Digital Image Processing Using Matlab*. Prentice-Hall, Englewood Cliffs (2003)
11. Avrahami, G., Pratt, V.: Sub-pixel edge detection in character digitization. *Raster Imaging and Digital Typography II*, pp. 54–64 (1991)

12. Hou, Z.J., Wei, G.W.: A new approach to edge detection. *Pattern Recognition* 35, 1559–1570 (2002)
13. Sait, M.S., Youssef, H.: *Iterative Computer Algorithms with Applications in Engineering: Solving Combinatorial Optimization Problems*. IEEE Computer Society Press, California (1999)
14. Masood, A., Sarfraz, M.: Capturing Outlines of 2D Objects with Bézier Cubic Approximation. *International Journal of Image and Vision Computing* 27(6), 704–712 (2009)

Part VI

**Special Session Soft Computing for
Intelligent Control**

Embedding a KM Type Reducer for High Speed Fuzzy Controller into an FPGA

Roberto Sepúlveda, Oscar Montiel-Ross, Oscar Castillo, and Patricia Melin

Abstract. There are many research works that have shown the advantages of type-2 fuzzy inference systems (T2-FIS) handling uncertainty with respect to type-1 fuzzy inference systems (T1-FIS); however, the use of a T2-FIS is still being controversial for several reasons, one of the most important is related to the resulting shocking increase in computational complexity that type reducers cause even for small systems, for example the Karnik-Mendel (KM) iterative method. The main goal of this paper is to show that the KM type reducer can be an efficient method if it is adequately implemented using the appropriate combination of hardware and software. In this work a novel architecture to implement the KM type reducer is shown, and in order to evaluate the architecture a comparative study was conducted. The study consisted in using a type-2 FIS programmed in Matlab to obtain some benchmarks, this to contrast the obtained results by testing the FIS programmed in VHDL for FPGA implementation. Preliminary studies have shown that the resulting speed up is in the order of 10^3 , since a typical whole T2-inference (fuzzification, inference, KM-type reducer, and defuzzification) last 5 clock cycles; i.e., 0.1×10^{-6} seconds for a Spartan 3 FPGA based system. Comparisons of the resulting control surfaces between T2-FIS programmed in Matlab and the FPGA implementation are also presented.

1 Introduction

Recently, there has been an increasing interest in the research and implementations of type-2 fuzzy systems because they offer bigger advantages in handling

Roberto Sepúlveda · Oscar Montiel Ross

Instituto Politécnico Nacional, Centro de Investigación y Desarrollo de Tecnología Digital (CITEDI-IPN), Av. del Parque No.1310, Mesa de Otay, 22510, Tijuana, B.C., México
e-mail: rsepulve@citedi.mx, o.montiel@ieee.org

Oscar Castillo · Patricia Melin

Tijuana Institute of Technology, Tijuana, B.C., México

e-mail: ocastillo@hafsamx.org, pmelin@tectijuana.edu.mx

uncertainty with respect to type-1 fuzzy systems [2, 3, 6, 7, 11, 12, 19]; many researchers have found this characteristic very interesting to be applied in the design of controllers. This statement is supported by several research reports that cover from theoretical studies to physical implementations [20, 21]. Most of the existing implementations of type-2 fuzzy systems have been achieved in software on general-purpose computers, the main drawback of these implementations is the speed limitation due to the sequential computer program execution [22]. An option for the implementation of fuzzy systems that require high processing speed to operate in real-time is based on dedicated hardware, so in order to achieve high performance systems, the use of VLSI devices is increasing [18].

There are other works related to this one, in [1] an embedded type-2 fuzzy controller based on the Texas Instruments TMS320F2812 150 MHz industrial DSP programmed in ANSI is described, used to control a 3 phase brushless DC motor via a 50W power inverter. In this case the Wu-Mendel [3] method was applied as a good approximation to the iterative Karnik-Mendel method.

With regards to a type-2 FIS in FPGA, in [16] the implementation of the Wu-Mendel method in a XC2V3000ff1152-4 FPGA type of the Virtex family is presented, taking advantages of the intrinsic parallelism that these devices offer; also we can find some others works related on this topic in [15, 17].

This paper is about the design, test and implementation of an interval type-2 fuzzy system based on the Karnik-Mendel type reducer [7]. It describes the design and implementation in FPGA of a type-2 Fuzzy Inference Systems (FIS) using the Karnik-Mendel type reducer. The fuzzification block has two inputs, and one output. In this stage the proposed architecture uses only active Membership Functions (MFs), an efficient idea to achieve decimal floating-point for decimal encoding, as well for binary encoding is used. An improved high performance type-2 inference engine (IE) is presented, which is divided in two entities the upper IE and the lower IE. An interesting proposal for the type reducer, which takes advantage of the inherent parallel features of the FPGAs, is explained. The proposed algorithms for the different stages were tested using the Xilinx System Generator (XSG) from Xilinx [14], and the Matlab/Simulink from Mathworks.

This paper has been organized as follows: Section 2 is devoted to give an introduction of type-2 FIS, some useful definitions about the mathematics for interval type-2 FIS are given. In section 3 the hardware implementation of a type-2 FIS is presented, in section 3.1 the fuzzification stage is explained. In Section 3.2 the type-2 Inference Engine stage designed using the IEs is explained, as well as its VHDL implementation. In section 3.3 the design and implementation of the KM type reducer is presented. In section 3.4 the defuzzification and its implementation is explained. Section 4 was dedicated to explain the general idea that was followed in testing and validating the interval type-2 FIS with the KM type reducer. Moreover, a Simulink model that allows to test the developed type-2 VHDL code for the type-2 FIS is shown. Finally, in section 5 some conclusions about this work are presented.

2 General Concepts of Type-2 FIS

The basics and principles of fuzzy logic do not change from type-1 to type-2 fuzzy sets [4, 5, 6], they are independent of the nature of membership functions, and in general, will not change for any type-n. When a FIS uses at least one type-2 fuzzy set is a type-2 FIS, which is shown in Figure 1 with its components. The structure of the type-2 fuzzy rules, is the same as for the type-1 case because the distinction between them is associated with the nature of the membership functions. Hence, the only difference is that now some or all the sets involved in the rules are of type-2, and as long as any of its antecedents or consequents sets are interval type-2 fuzzy sets, it is called an interval type-2 FIS (IT2FIS) [7, 13] which is the most used. An IT2 FIS contains four components: fuzzification, inference engine, type-reducer, and defuzzification, which are connected as it is shown in Figure 1. The IT2 FIS can be seen as a mapping from the inputs to the output and it can be interpreted quantitatively as $Y = f(X)$, where $X = \{x_1, x_2, \dots, x_n\}$ are the inputs to the IT2 FIS f , and $Y = \{y_1, y_2, \dots, y_n\}$ are the defuzzified outputs. In a type-1 FIS, where the output sets are type-1 fuzzy sets, the defuzzification is performed to get a number, which is in some sense a crisp representation of the combined output sets. In the type-2 case, the output sets are type-2, so it is necessary the extended defuzzification operation to get type-1 fuzzy set at the output. Since this operation converts type-2 output sets to a type-1 fuzzy set, it is called type reduction, and the type-1 fuzzy set obtained is called a type-reduced set, the type-reduced fuzzy set may then be defuzzified to obtain a single crisp number.

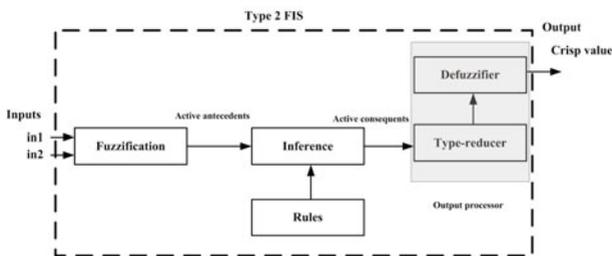


Fig. 1 Type-2 fuzzy system

3 Hardware Implementation of a Type-2 FIS

For explaining the IT2 FIS implementation, the rule base of a speed control of a DC motor was used, which is presented in Figure 2. The system has two inputs: “Error” and “Change of error”, with five T2 FS each, and one output also with five T2 FS.

Rule Base

Error \ Error	NB	N	Z	P	PB
NB	BI	BI	BI	BI	BI
N	BI	I	I	H	I
Z	I	H	H	H	D
P	D	H	D	D	BD
PB	BD	BD	BD	BD	BD

Fig. 2 Rule base for the speed control of a DC motor used for the system validation

3.1 Fuzzification Stage

The fuzzification stage has two input variables: “Error” and “Change of Error”. Each linguistic variable has five terms identified by linguistic tags, and each tag is related with three binary digits as follows, “Negative Big” (NB) with the value of “001”, “Negative” (N) with “010”, Zero (Z) with “011”, “Positive” with “100”, and Positive Big (PB) with “101”. For this architecture, the maximal number of active Type 2 membership functions for each input is two. Hence in Figure 3 at the input “Error” there is a crisp value of ≈ -25 that is activating the linguistic terms NB (“001”) and N (“010”). For this example, considering only the upper MFs of the FOU, the membership values are ≈ 0.66 and ≈ 0.33 , the upper active membership functions (linguistic terms) and upper membership values are assigned to VHDL variables, in the figure, $e_1Up = “001”$ (NB), $e_2Up = “010”$ (N), for the

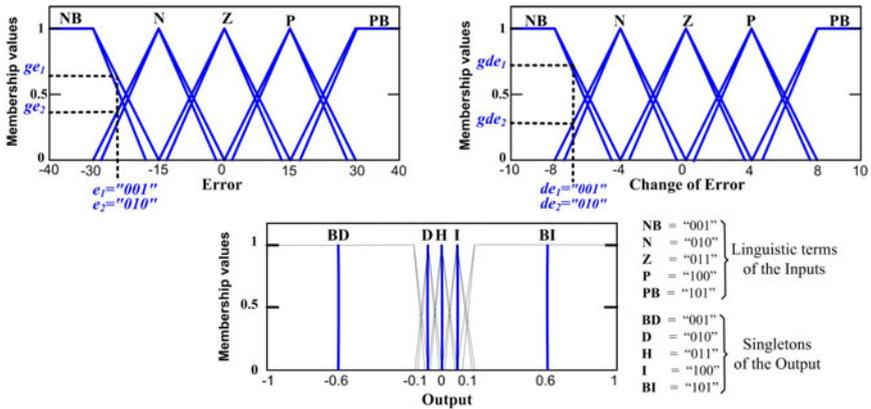


Fig. 3 Linguistic Variables for the inputs and output: Error, change of error, and output with their linguistic terms.

former, the upper membership value is $ge_1Up = 0.66$, and for the latter, $ge_2Up = 0.33$. Similarly, the second input “Change of Error” is handled, the linguistic terms are assigned to the VHDL variables de_1Up and de_2Up , and their upper membership values are gde_1Up and gde_2Up . Finally, the four VHDL variables of each input are assigned to VHDL output signals that are the inputs to the next stage; i.e. the IEEup. The fuzzification Stage Entity (FSE) was designed using the behavioral style of VHDL codification, this style use sequential code, and it only needs one clock pulse to perform the fuzzification.

Considering the upper and lower MFs of the FOU of each T2MF, there are four upper MFs with their respective grades of membership, as well as four lower MFs, also with their grades of membership at the output of the fuzzification stage. The block of this stage in the Xilinx System Generator is shown in Figure 6.

3.2 Inference Engine Stage

Figure 4 shows the interaction of the fuzzification stage, and the inference engine for an IT2 FIS. The main block of this picture labeled as “Inference Engine” is a VHDL entity, and it gives a general idea of its internal architecture. It is divided in two parallel Inference Engine Entities (IEEs), one of them is to manage the upper bound of the IT2FSs (IEEup) and the other one for the lower bound of the IT2FSs (IEElow). Each one has nine inputs and eight outputs, eight of inputs came from the fuzzification stage, the ninth input is a clock signal; with respect to the eight outputs, four are for the linguistic terms, the rest correspond to their firing strengths.

In order to explain the operation of the Inference stage, the IEEup block is described. At the input of the IEEup of Figure 5 are the linguistic terms (e_1Up , e_2Up , de_1Up , and de_2Up), their upper membership values (ge_1Up , ge_2Up , gde_1Up , and

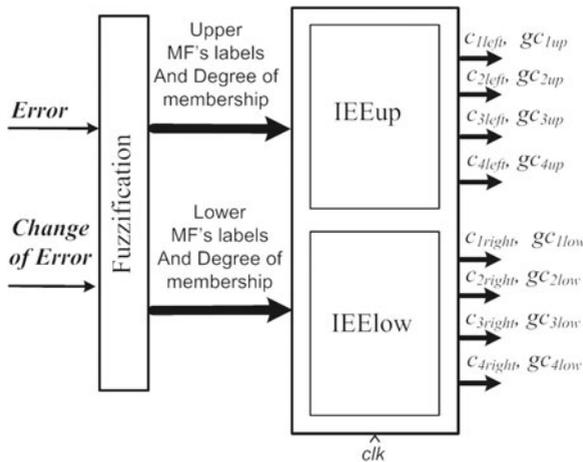


Fig. 4 Overview of the inference engine architecture

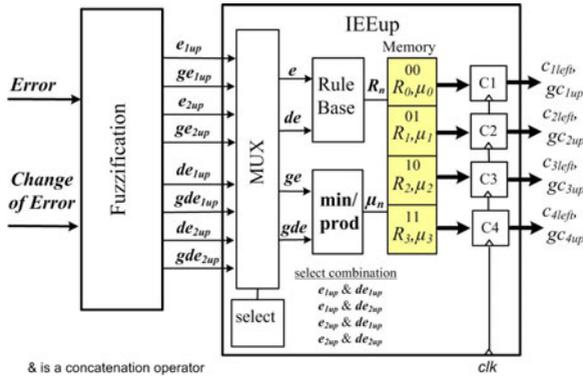


Fig. 5 Detail of the upper inference engine architecture

gde_2Up), and the clock signal (clk). All the inputs, except clk , enter into a parallel selection process through a multiplexer; note that although the IEEUp uses sequential code, the circuits into the process are placed in parallel; the degree of parallelism can be tailored by an adequate codification style, in our case all the rules are processed in parallel, the eight outputs are obtained at the same time because the “ clk ” signal synchronizes the process, hence this stage needs only one clock cycle to provide the output.

In the IEEUp, the required circuits to implement the MUX and Select blocks are placed in parallel, and they produce four outputs; by the concatenation of two of them, e and de the active rules are identified.

For example, for the combination of $e_1Up = “010”$ and $e_2Up = “010”$ the value of the variable “ante” is “010010” and the linked consequent is “I” with a binary value of “100”, as it can be seen in Figure 3. The common rule notation is shown next as rule number 4.

1. If e_1Up is “001” and de_1Up is “001” then C_1Left is BI
2. If e_1Up is “001” and de_2Up is “010” then C_2Left is BI
3. If e_2Up is “010” and de_1Up is “001” then C_3Left is BI
4. If e_2Up is “010” and de_2Up is “010” then C_4Left is I
5. ⋮

The other two outputs of the MUX block are the membership values ge and gde , the minimal t-norm of the corresponding rule is calculated to obtain the firing strength. The e_1Up and de_1Up combination and its linked consequent are saved into VHDL variables, this has been illustrated in Figure 5 as the memory position “00” that contains the consequent stored in R_0 and its firing strength stored in μ_0 ; the rule combination e_1Up and de_2Up and its consequent are stored in R_1 and in μ_1 , in the memory position “01”, etc.

For example, if the memory position “00” has the consequent of the active rule 1, i.e. the combination ‘001001’, then the consequent is BI (“101”); if the memory

position “01” has the consequent of the active rule 2, the combination ‘001010”, then the consequent is BI; if the memory position “10” has the consequent of the active rule 3, the consequent of the rule combination “010001” is also BI, and so forth.

The “Memory” block in Figure 5 at the output is connected to four blocks named C_1 to C_4 , each C_x block represents two VHDL signals where the triggered consequent C_xLeft and its membership value or firing strength gc_xLeft are saved and sent to the entity output at the next clock cycle. Note that this method only needs four C_x blocks because the maximal number of active rules is four, and that the maximum required time to perform an inference is only one clock cycle.

Figure 6 shows the IEE that was produced by the Xilinx ISE software as result of the synthesis process. This entity shows two extra inputs, “Chip Enable” (ce) and “Reset” (rst), they were added only to be compatible with the Xilinx System Generator (XSG) and Simulink.

3.3 Type Reducer Stage

At the input of the Type Reducer we have the equivalent values of the precomputed y_l^i , i.e. the linguistic terms of the active consequents (C_1Left , C_2Left , C_3Left , and C_4Left), the upper firing strength ($gc1Up$, $gc2Up$, $gc3Up$, and $gc4Up$); also the equivalent values of the precomputed y_r^i (C_1Right , C_2Right , C_3Right , and C_4Right), the lower firing strength ($gc1Low$, $gc2Low$, $gc3Low$, and $gc4Low$). All the above mentioned signals, except the reset signal (rst), enter into a parallel selection process to perform the KM algorithm [8]. There are parallel blocks to obtain the average of the upper and lower firing strength for the active consequents, required to obtain the average of the y_r and y_l , step two of the KM algorithm, a block to obtain the different defuzzified values of y_r and y_l , obtained according to step four of the KM method; parallel comparator blocks to obtain the final result of y_r and y_l , step five of the KM algorithm described in [8].

Figure 6 shows the Type Reducer that was produced by the Xilinx ISE software as result of the synthesis process. It is observed that there are 16 inputs and two outputs: the y_r and y_l . This entity shows also two extra inputs, “Chip Enable” (ce) and “Reset” (rst), to be compatible with the Xilinx System Generator (XSG) and Simulink.

3.4 Defuzzification Stage

The final result of the T2FIS is obtained using the defuzzification block, which computes the average of the y_r and y_l . So, this block has only two inputs and one output y .

In Figure 6 is shown the Deffuzzification Stage that was produced by the Xilinx ISE.

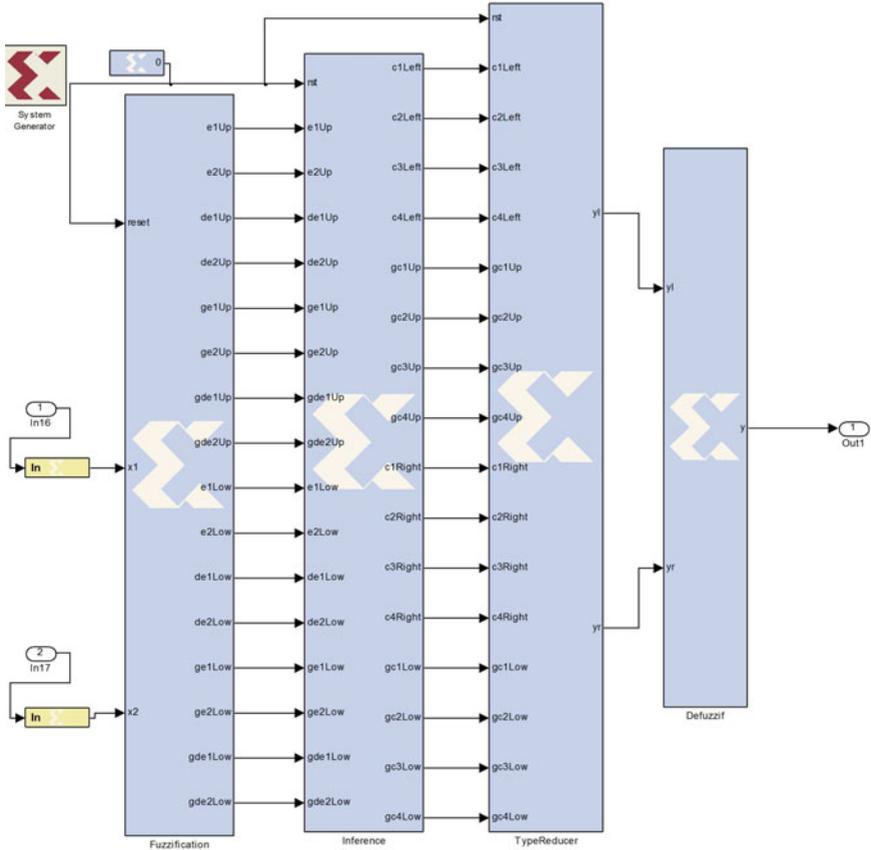


Fig. 6 Entity of the type-2 FLS engine

4 Testing of the Type 2 FIS and Results

In order to test the complete T2 FIS, shown in Figure 6, three main different software tools were used: The Simulink from Mathworks which is a practical high-level design and simulation tool because it provides a flexible design and simulation platform to test and correct designs at high level; the Xilinx Integrated Software Environment (Xilinx ISE), which is a Hardware Description Language (HDL) design software suite, that allows taking designs through several steps in the ISE design flow, finishing with final verified modules that can be implemented in a hardware target, such as a Field Programmable Gate Array (FPGA); and the Xilinx System Generator (XSG), which is a DSP design tool that enables the use of the Simulink for FPGA design. This tool is very important because it allows generating VHDL code from the System Generator Simulink modules; and viceversa, VHDL modules can be included in the Simulink design platform by importing the VHDL code in a System Generator “Black box”, the VHDL code is converted to Matlab functions. There

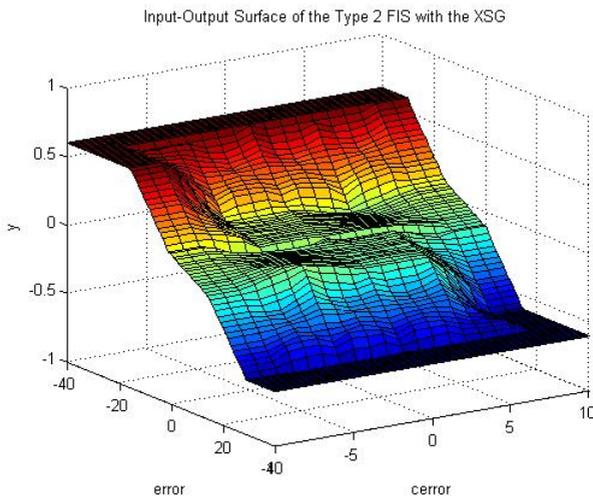


Fig. 7 Control Surface with the T2 FIS proposal

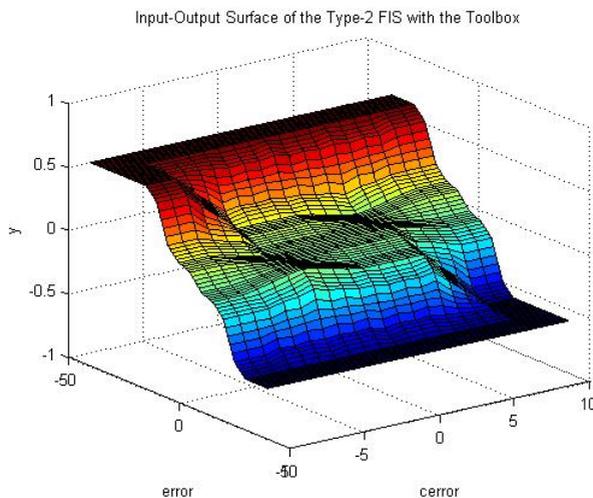


Fig. 8 Control Surface obtained with the Type-2 Fuzzy Logic Toolbox

were applied two vectors to the inputs of the T2 FIS, which include the possible values within the universe of discourse of the two inputs to the FIS. The control surface obtained is shown in Figure 7. The same T2 FIS using the Interval Type-2 Fuzzy Logic Toolbox in MATLAB software was designed [9, 10] and the control surface was also obtained, which is shown in Figure 8, all under the same conditions applied to our proposed T2 FIS in FPGA.

4.1 Comparison of Results

In Figure 9 the output error between the values obtained with the T2 FIS in FPGA and the one obtained with the type-2 Toolbox is shown. Those differences are caused because in our proposed method we used eight bits of resolution, whereas the system designed in the computer with the type-2 toolbox uses floating point number in the computations.

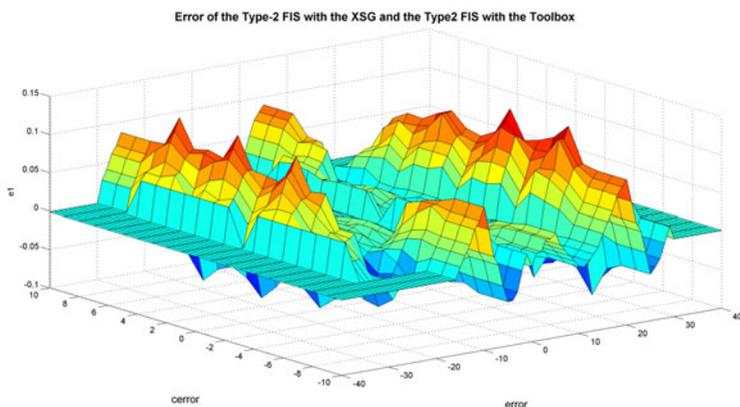


Fig. 9 Error between the control surface of type-2 FIS proposal and the control surface obtained with the type-2 fuzzy logic toolbox

5 Conclusions

The most widely adopted way to implement a fuzzy system is to use software into a non-dedicated computer. This method has the advantage of giving high level support to the designer since there are many friendly specializing software that helps to obtain results in short time, however one of the disadvantages of this solution is the processing time. Hardware implementations, specifically those designs that have been carry out to be implemented at circuitry level, offer bigger advantages to designers to provide high speed solutions.

In this paper, an improved type-2 inference engine was proposed for hardware implementation, oriented to be implemented into an FPGA. The preliminary results have shown that the resulting speed up is in the order of 10^3 , since a typical whole T2-inference (fuzzification, inference, KM-type reducer, and defuzzification) last 5 clock cycles; i.e., 0.1×10^{-6} seconds for a Spartan 3 FPGA based system. Comparison of the resulting control surfaces between T2-FIS programmed in Matlab and the FPGA implementation were shown.

A Simulink model was presented for testing the interval type-2 FIS with the KM in a FPGA, compared with a type-2 FIS designed with the type-2 Toolbox. The

control surfaces of both showed small differences, basically because eight bits of resolution was used in the former compared to the floating point numbers used in the computations in the late one. We conclude that the interval type-2 FIS proposal using the KM method, is suitable to implement high speed real time systems.

References

1. Lynch, C., Hagrass, H., Callaghan, V.: Embedded Type-2 FLC for Real-Time Speed Control of Marine and Traction Diesel engines. In: IEEE International Conference on Fuzzy Systems, pp. 347–352 (2005)
2. Hagrass, H.A.: Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots. IEEE Transactions on Fuzzy Systems 12(4), 524–539
3. Wu, H., Mendel, J.M.: Uncertainty Bounds and Their use in the Design of Interval type-2 Fuzzy Logic Systems. IEEE Transactions on Fuzzy Systems 10, 1–16 (2002)
4. Mendel, J.M.: Type-2 Fuzzy Sets and Systems: an Overview. IEEE Computational Intelligence Magazine 2, 20–29 (2007)
5. Mendel, J.M.: Type-2 Fuzzy Sets: Some Questions and Answers. IEEE Connections, Newsletter of the IEEE Neural Networks Society 1, 10–13 (2003)
6. Mendel, J.M., Bob John, R.I.: Type-2 Fuzzy Sets Made Simple. IEEE Transactions on Fuzzy Systems 10, 117–127 (2002)
7. Mendel, J.M.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice-Hall, Uppersaddle River (2001)
8. Mendel, J.M., Hagrass, H., John, R.I.: Standard Background Material About Interval Type-2 Fuzzy Logic Systems That Can Be Used By All Authors, http://iee-cis.org/_files/standards.t2.win.pdf
9. Castro, J.R., Castillo, O., Melín, P., Rodríguez, A.: Building Fuzzy Inference Systems with a new Interval Type-2 Fuzzy Logic Toolbox. In: Gavrilova, M.L., Tan, C.J.K. (eds.) Transactions on Computational Science I. LNCS, vol. 4750, pp. 104–114. Springer, Heidelberg (2008)
10. Castro, J.R., Castillo, O., Melín, P.: An interval type-2 fuzzy logic toolbox for control applications. In: Castro, J.R., Castillo, O., Melin, P. (eds.) 2008 Proceedings of FUZZ-IEEE, London, pp. 61–67 (2007)
11. Karnik, N.N., Mendel, J.M., Liang, Q.: Type-2 Fuzzy Logic Systems. IEEE Transactions on Fuzzy Systems 7, 643–658 (1999)
12. Karnik, N., Mendel, J.M.: Centroid of a Type-2 Fuzzy Set. Information Sciences 132, 195–220 (2001)
13. Liang, Q., Mendel, J.M.: Interval Type-2 Fuzzy Logic Systems: Theory and Design. IEEE Trans. on Fuzzy Systems 8, 535–550 (2000)
14. Manual of Xilinx System Generator, <http://www.xilinx.com>
15. Melgarejo, M., Peña-Reyes, C.A.: Implementing Interval type-2 Fuzzy Processors. IEEE Computational Intelligence Magazine 2, 63–71 (2007)
16. Melgarejo, M.A., Garcia, R.A., Peña-Reyes, C.A.: Hardware architecture and FPGA implementation of a type-2 fuzzy system, Proceedings of the 14th ACM Great Lakes symposium on VLSI, pp. 458–461 (2004)
17. Melgarejo, M.A., Garcia, R.A., Peña-Reyes, C.A.: Pro-Two: a hardware based platform for real time type-2 fuzzy inference. In: Proceedings of 2004 IEEE International Conference on Publication Fuzzy Systems, vol. 2, pp. 977–982 (2004)
18. Cirstea, M.N., Dinu, A., Khor, J.G., McCormick, M.: Neural and Fuzzy Logic Control of Drives and Power System, Newnes (2002)

19. Melin, P., Castillo, O.: A New Method for Adaptive Control of Non-Linear Plants using Type-2 Fuzzy Logic and Neural Networks. *International J. of General Systems* 33(2), 289–304 (2004)
20. Sepúlveda, R., Castillo, O., Melin, P., Díaz, A.R., Montiel, O.: Experimental Study of Intelligent Controllers under Uncertainty using type-1 and type-2 Fuzzy Logic. *Information Sciences* 177, 2023–2048 (2007)
21. Sepúlveda, R., Castillo, O., Melín, P., Montiel, O.: An Efficient Computational Method to implement Type-2 Fuzzy Logic in Control Applications. In: *Analysis and Design of Intelligent Systems Using Soft Computing Techniques, Advances in soft computing*, vol. 41, pp. 45–52. Springer, Heidelberg (2007)
22. Poorani, S., Urmila Priya, T.V.S.: FPGA Based Fuzzy Logic Controller for Electric Vehicle. *Journal of the Institution of Engineers* 45(5), 1–14 (2005)

An Application of Fuzzy Lyapunov Synthesis in the Design of Type-2 Fuzzy Logic Controllers

Nohe R. Cazarez-Castro, Luis T. Aguilar, and Oscar Castillo

Abstract. The problem is the design of a feedback controller so as to obtain the closed-loop system in which all trajectories are bounded and the load of the driver is regulated to a desired position while also attenuating the influence of external disturbances. This problem is solved by extending the Fuzzy Lyapunov System to the design of Type-2 Fuzzy Controllers for a servomechanism with nonlinear backlash. The servomotor position is the only measurement available for feedback; the proposed extension is far from trivial because of nonminimum phase properties of the system. Performance issues of the Type-2 Fuzzy Regulator designed are illustrated in a simulation study.

Keywords: type-2 fuzzy control, fuzzy Lyapunov synthesis, backlash.

1 Introduction

A major problem in control engineering is a robust feedback design that asymptotically stabilizes a nonminimal plant while also attenuating the influence of parameter variations and external disturbances. In the last decade, this problem was heavily studied and considerable research efforts have resulted in the development of systematic design methodology for nonlinear feedback systems (see [8]).

Nohe R. Cazarez-Castro

Universidad Autonoma de Baja California, Calzada Universidad 14418, Parque Industrial Internacional Tijuana, B.C., Mexico, 22414
e-mail: nohe@ieee.org

Luis T. Aguilar

Instituto Politecnico Nacional, Mexico
e-mail: luis.aguilar@ieee.org

Oscar Castillo

Instituto Tecnológico de Tijuana, Mexico
e-mail: ocastillo@hafsamx.org

In the present paper the output regulator problem is studied for an electrical actuator consisting of a motor part driven by a DC motor and a reducer part (load) operating under uncertainty conditions in the presence of nonlinear backlash effects. The objective is to drive the load to a desired position while providing the boundedness of the system motion and attenuating external disturbances. Because of practical requirements [9], the motor angular position is assumed to be the only information available for feedback. This problem is present in many industrial applications, for example, manipulator robots working in hazard environments, in such applications is very difficult (even impossible) to place a sensor for position feedback of the final effector of the manipulator robot, leaving the motor's position as the only source of feedback to make decisions on control actions.

Since the Type-2 Fuzzy Logic Control pioneers papers [6] and [11], has been a large research effort in this area, in [7] there is a wide review and references. The problem that we presents in this paper was first reported in [1], where the nonminimum phase systems control problem was solved by using nonlinear H_∞ control, but the reported results do not provide robustness evidence. In [5], authors report a solution to the regulation problem by using a type-1 fuzzy logic controller. In [3] were reported solutions by using type-2 fuzzy logic controllers, and making a genetic optimization of the membership function's parameters, but do not specify the criteria used in the optimization process and Genetic Algorithm (GA) design. In [4], a comparison of the use of GA to optimize type-1 and type-2 fuzzy logic controllers is reported, but a method to achieve this optimization is not provided.

The solution that we propose in this paper is to design a type-2 fuzzy controller extending the Fuzzy Lyapunov Synthesis [10], a concept that is based on the Computing with Words approach of the Lyapunov Synthesis, this approach was reported in [2] for the design of stable type-2 fuzzy logic controllers.

The paper is organized as follows. The dynamic model of the servomechanism is presented in Section 2. The problem statement is in Section 3. Section 4 address Type-2 Fuzzy Sets and Systems theory. The design of the Type-2 Fuzzy Controller is described in Section 5. The numerical simulations for the designed Type-2 Fuzzy Controller are presented in Section 6. Conclusions are presented in Section 7.

2 Dynamic Model

The dynamic model of the angular position $q_i(t)$ of the DC motor and $q_o(t)$ of the load are given according to

$$\begin{aligned} J_0 N^{-1} \ddot{q}_0 + f_0 N^{-1} \dot{q}_0 &= T + w_0 \\ J_i \ddot{q}_i + f_i \dot{q}_i + T &= \tau_m + w_i, \end{aligned} \quad (1)$$

hereafter, J_0 , f_0 , \ddot{q}_0 and \dot{q}_0 are, respectively, the inertia of the load and the reducer, the viscous output friction, the output acceleration, and the output velocity. The inertia of the motor, the viscous motor friction, the motor acceleration, and the motor velocity denoted by J_i , f_i , \ddot{q}_i , and \dot{q}_i , respectively. The input torque τ_m serves as a

control action, and T stands for the transmitted torque. The external disturbances $w_i(t), w_0(t)$ have been introduced into the driver equation (1) to account for destabilizing model discrepancies due to hard-to-model nonlinear phenomena, such as friction and backlash.

The transmitted torque T through a backlash with an amplitude j is typically modeled by a dead-zone characteristic [14, p. 7]:

$$T(\Delta q) = \begin{cases} 0 & |\Delta q| \leq j \\ K\Delta q - Kj\text{sign}(\Delta q) & \text{otherwise} \end{cases} \quad (2)$$

with $\Delta q = q_i - Nq_0$, where K is the stiffness, and N is the reducer ratio. Provided the servomotor position $q_i(t)$ is the only available measurement on the system, the above model appears to be non-minimum phase because along with the origin the unforced system possesses a multivalued set of equilibria (q_i, q_0) with $q_i = 0$ and $q_0 \in [-j, j]$.

To avoid dealing with non-minimum phase system, we replace the backlash model (2) with its monotonic approximation:

$$T = K\Delta q - K\eta(\Delta q) \quad (3)$$

where

$$\eta = -2j \frac{1 - \exp\left\{-\frac{\Delta q}{j}\right\}}{1 + \exp\left\{-\frac{\Delta q}{j}\right\}}. \quad (4)$$

3 Problem Statement

To formally state the problem, let us introduce the state deviation vector $x = [x_1, x_2, x_3, x_4]^T$ with

$$\begin{aligned} x_1 &= q_0 - q_d, & x_3 &= q_i - Nq_d, \\ x_2 &= \dot{q}_0, & x_4 &= \dot{q}_i, \end{aligned}$$

where x_1 is the load position error, x_2 is the load velocity, x_3 is the motor position deviation from its nominal value, and x_4 is the motor velocity. The nominal motor position Nq_d has been pre-specified in such a way to guarantee that $\Delta q = \Delta x$, where $\Delta x = x_3 - Nx_1$. Then, system (1)–(4), represented in terms of the deviation vector x , takes the form

$$\begin{aligned} \dot{x}_1 &= x_2, & \dot{x}_2 &= J_0^{-1}[KNx_3 - KN^2x_1 - f_0x_2 + KN\eta(\Delta q) + w_0], \\ \dot{x}_3 &= x_4, & \dot{x}_4 &= J_i^{-1}[\tau_m + KNx_1 - Kx_3 - f_ix_4 + K\eta(\Delta q) + w_i]. \end{aligned} \quad (5)$$

The zero dynamics

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = J_0^{-1}[-KN^2x_1 - f_0x_2 + KN\eta(-Nx_1)], \quad (6)$$

of the undisturbed version of system (5) with respect to the output $y = x_3$ is formally obtained by specifying the control law that maintains the output identically to zero.

The objective of the Fuzzy Control output regulation of the nonlinear driver system (1) with backlash (3) and (4), is thus to design a Fuzzy Controller so as to obtain the closed-loop system in which all these trajectories are bounded and the output $q_0(t)$ asymptotically decays to a desired position q_d as $t \rightarrow \infty$ while also attenuating the influence of the external disturbances $w_i(t)$ and $w_0(t)$.

4 Type-2 Fuzzy Sets and Systems

The concept of Type-2 Fuzzy Set was introduced by Zadeh as an extension of the concept of an ordinary Fuzzy Set. A Type-2 Fuzzy Set, denoted as \tilde{A} is characterized by a Type-2 Membership Function $\mu_{\tilde{A}}(z, \mu(z))$ [13], where $z \in Z$ and $\mu \in J_z \subseteq [0, 1]$, i.e.,

$$\tilde{A} = \{((z, \mu(z)), \mu_{\tilde{A}}(z, \mu(z))) | \forall z \in Z, \forall \mu(z) \in J_z \subseteq [0, 1]\} \quad (7)$$

in which $0 \leq \mu_{\tilde{A}}(z, \mu(z)) \leq 1$.

J_z is called primary membership of z , where $J_z \subseteq [0, 1]$ for $\forall z \in Z$ [13]. The uncertainty in the primary memberships of a Type-2 Fuzzy Set \tilde{A} , consists of a bounded region that is called the *footprint of uncertainty* (FOU) [13]. It is the union of all primary memberships [13].

An Interval Type-2 Fuzzy Set \tilde{A} is to date the most widely used kind of Type-2 Fuzzy Sets, and is the only kind of Type-2 Fuzzy Sets that is considered in this paper. It is described as

$$\tilde{A} = \int_{z \in Z} \int_{\mu(z) \in J_z} 1/(z, \mu(z)) = \int_{z \in Z} \left[\int_{\mu(z) \in J_z} 1/\mu(z) \right] /z, \quad (8)$$

where z is the *primary variable*, $J_z \subseteq [0, 1]$ is the *primary membership* of z , $\mu(z)$ is the *secondary variable*, and $\int_{\mu(z) \in J_z} 1/\mu(z)$ is the *secondary membership function* at z . Note that (8) means $\tilde{A} : Z \rightarrow \{[a, b] : 0 \leq a \leq b \leq 1\}$. Uncertainty about \tilde{A} is conveyed by the union of all of the primary memberships, called the FOU of \tilde{A} [FOU(\tilde{A})], i.e.,

$$\text{FOU}(\tilde{A}) = \bigcup_{z \in X} J_z. \quad (9)$$

A Fuzzy Logic System described using at least one Type-2 Fuzzy Set is called a Type-2 Fuzzy Logic Systems - also called Type-2 Fuzzy Inference Systems -. Type-1 Fuzzy Inference Systems are unable to directly handle rule uncertainties, this is because they use Type-1 Fuzzy Sets that are certain. On the other hand, Type-2 Fuzzy Inference Systems are very useful in circumstances where it is difficult to determine an exact and measurement uncertainties [12].

Similar to a Type-1 Fuzzy Inference Systems, a Type-2 Fuzzy Inference System includes *Type-2 fuzzifier*, *Type-2 rule-base*, *Type-2 inference engine* and substitutes the *Type-1 defuzzifier* by the *output processor*. The *output processor* includes a *type-reducer* [12] and a *Type-2 defuzzifier*; it generates a Type-1 Fuzzy

Set output (from the *type-reducer*), or a crisp number (from the *Type-2 defuzzifier*). A Type-2 Fuzzy Inference System is again characterized by IF-THEN rules, but its antecedent and consequents sets are now of the Type-2, see (10). Type-2 Fuzzy Inference Systems can be used when the circumstances are too uncertain to determine exact membership grades.

$$R^l : \text{IF } y \text{ is } \tilde{A}_1^l \text{ AND } \dot{y} \text{ is } \tilde{A}_2^l \text{ THEN } u \text{ is } \tilde{B}_1^l, \tag{10}$$

In this paper we are computing a Centroid type-reducer (CTR) (12) as *Type-2 defuzzifier*, the CTR is defined as follows:

$$\tau_m \int_{\theta_j \in J_N} \cdots \int_{\theta_N \in J_{u_N}} [f_{u_1}(\theta_1) * \cdots * f_{u_N}(\theta_N)] / \frac{\int u_i \theta_i d\theta}{\int \theta_i d\theta}. \tag{11}$$

5 Design of the Type-2 Fuzzy Logic Controller

To apply the Fuzzy Lyapunov Synthesis method, we assume that the exact equations are unknown and that we have only the following partial knowledge about the plant:

1. The system may have really two degrees of freedom referred to as x_1 and x_2 , respectively. Hence by (5), $\dot{x}_1 = x_2$.
2. \dot{x}_2 is proportional to τ_m , that is, when τ_m increases (decreases) \dot{x}_2 increases (decreases).

Our objective is to design the rule-base of a Type-2 Fuzzy Controller that will carry the load to a desired position q_d , or in other words, that will carry the trajectories of x_1 to zero. Define the Lyapunov function:

$$V(x_1, x_2) = \frac{1}{2} (x_1^2 + x_2^2) \tag{12}$$

which is positive definite. The time derivative of $V(x_1, x_2)$ is

$$\dot{V}(x_1, x_2) = x_1 \dot{x}_1 + x_2 \dot{x}_2 = x_1 x_2 + x_2 \dot{x}_2, \tag{13}$$

hence, we require that

$$x_1 x_2 + x_2 \dot{x}_2 \leq 0. \tag{14}$$

We can now derive sufficient conditions so that condition (14) holds: If x_1 and x_2 have opposite signs, then $x_1 x_2 < 0$ and (14) will hold if $\dot{x}_2 = 0$; if x_1 and x_2 are both positive, then (14) will hold if $\dot{x}_2 < -x_1$; if x_1 and x_2 are both negative, then (14) will hold if $\dot{x}_2 > -x_1$.

We can translate these conditions into the following fuzzy rules:

- If x_1 is *positive* and x_2 is *positive* then u must be *negative big*.
- If x_1 is *negative* and x_2 is *negative* then u must be *positive big*.
- If x_1 is *positive* and x_2 is *negative* then u must be *zero*.
- If x_1 is *negative* and x_2 is *positive* then u must be *zero*.

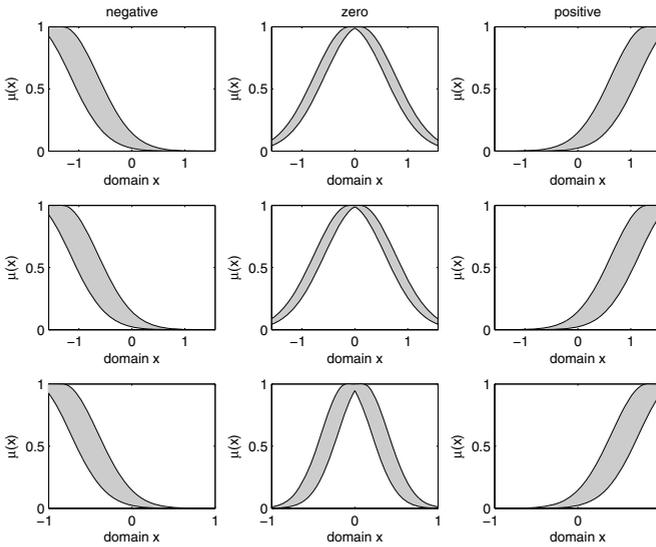


Fig. 1 Set of type-2 membership functions

To complete the controllers design, we must model the linguistic terms in the rule-base using fuzzy membership functions and determine an inference method. Following [2], we characterize the linguistic terms *positive*, *negative*, *negative big*, *zero* and *positive big* by the type-2 membership functions shown in Fig. 1 for a Type-2 Fuzzy Logic Controller, note that the membership function *zero* is not depicted for error and change of error variables. To this end, we had systematically design a Type-2 Fuzzy Logic Controller following the Lyapunov stability criterion.

6 Simulation Results

To perform simulations we use the dynamical model (1)–(4), which involves a DC motor linked to a mechanical load through an imperfect contact gear train (1). The parameters of the dynamical model (1) are Motor inertia $J_i 2.8 \times 10^{-6}$ Kg-m², Load inertia $J_o = 1.07$ Kg-m², Motor viscous friction $f_i = 7.6 \times 10^{-7}$ N-m-s/rad, Load viscous friction $f_o = 1.73$ N-m-s/rad, while $N = 3$, $j = 0.2$ [rad], and $K = 5$ [N-m/rad].

Performing a simulation of the closed-loop system (1)–(3) with the Type-2 Fuzzy Logic Controller designed in Section 5 we have the following results: the surface of control of Fig. 2 and the system output of Fig. 3. It can be seen that load trajectories reach the desired position as was predicted. That is x_1 converges to zero, it can be seen in Fig. 3a, where you can see that the error stabilizes between seconds 10 and 14, at the same time that the change of the error is oscillating, this is due to the mechanism’s inertia, the fuzzy system does not provide a strong control signal but the mechanism continues in motion due to its inertia moment.

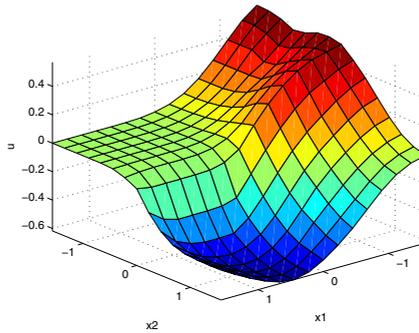


Fig. 2 Surface of control

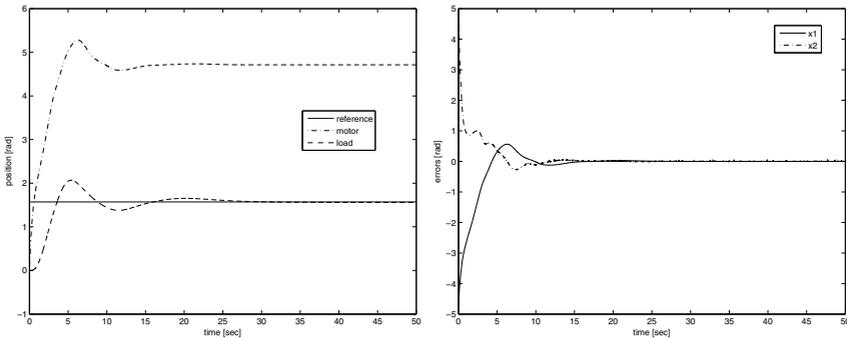


Fig. 3 a)Simulation result. b)Convergence of x_1 to zero.

7 Conclusion

Following the Fuzzy Lyapunov Synthesis we design a Type-2 Fuzzy Logic Controller, this controller was used to solve de control output regulation problem of a servomechanism with nonlinear backlash.

The proposed design strategy results in a controller that as was predicted, guarantees that the load reaches the desired position q_d , and the trajectories of the error x_1 converge to zero.

Type-2 Fuzzy Logic Systems were proved to be an appropriate control strategy for applications where not all the variables are available for measurement, this control strategy can save costs in industrial applications.

References

1. Aguilar, L.T., Orlov, Y., Cadiou, J.C., Merzouki, R.: Nonlinear H_∞ -output regulation of a nonminimum phase servomechanism with backlash. *Journal of Dynamic Systems, Measurement, and Control* 129(4), 544–549 (2007), <http://link.aip.org/link/?JDS/129/544/1>
2. Castillo, O., Aguilar, L.T., Cazarez, N., Cardenas, S.: Systematic design of a stable type-2 fuzzy logic controller. *Journal of Applied Soft Computing* 8(3), 1274–1279 (2008)
3. Cazarez-Castro, N., Aguilar, L., Castillo, O.: Hybrid genetic-fuzzy optimization of a type-2 fuzzy logic controller. In: Eighth International Conference on Hybrid Intelligent Systems, HIS 2008, pp. 216–221 (2008)
4. Cazarez-Castro, N., Aguilar, L., Castillo, O., Rodriguez, A.: Optimizing type-1 and type-2 fuzzy logic systems with genetic algorithms. *Research in Computing Science* 39, 131–153 (2008)
5. Cazarez-Castro, N.R., Aguilar, L.T., Castillo, O., Cardenas, S.: Fuzzy Control for Output Regulation of a Servomechanism with Backlash. In: *Soft Computing for Hybrid Intelligent Systems*, 1st edn. *Studies in Computational Intelligence*, vol. 154, pp. 19–28. Springer, Berlin (2008)
6. Hagrass, H.: A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots. *IEEE Transactions on Fuzzy Systems* 12(4), 524–539 (2004)
7. Hagrass, H.: Type-2 flcs: A new generation of fuzzy controllers. *IEEE Computational Intelligence Magazine* 2(1), 30–43 (2007)
8. Isidori, A.: A tool for semi-global stabilization of uncertain non-minimum-phase non-linear systems via output feedback. *IEEE Transactions on Automatic Control* 45(10), 1817–1827 (2000)
9. Lagerberg, A., Egardt, B.: Estimation of backlash with application to automotive powertrains. In: *Proc. of the 42nd Conf. on Decision and Control*, pp. 135–147 (1999)
10. Margaliot, M., Langholz, G.: Adaptive fuzzy controller design via fuzzy lyapunov synthesis. In: *The 1998 IEEE International Conference on Fuzzy Systems Proceedings, IEEE World Congress on Computational Intelligence*, vol. 1, pp. 354–359 (1998)
11. Melin, P., Castillo, O.: A new method for adaptive control of non-linear plants using type-2 fuzzy logic and neural networks. *International Journal of General Systems* 33(2&3), 289–304 (2004)
12. Mendel, J.: *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Prentice Hall, Upper Saddle River (2001)
13. Mendel, J., John, R.: Type-2 fuzzy sets made simple. *IEEE Transactions on Fuzzy Systems* 10(2), 117–127 (2002)
14. Nordin, M., Bodin, P., Gutman, P.: *New Models and Identification Methods for Backlash and Gear Play*. In: *Adaptive Control of Nonsmooth Dynamic Systems*, pp. 1–30. Springer, Berlin (2001)

Access Control and Resource Allocation for Peer-to-Peer Video Streaming Based on Fuzzy Logic

Zhengping Wu and Hao Wu

Abstract. The fast development of peer-to-peer video-on-demand (P2P-VoD) technologies provides people new experience of watching videos and live TVs online. However, reliability of the streaming source and resource allocation are still concerns from end-users. This paper introduces an adaptive control framework using fuzzy logic into user preference definition for peer-to-peer (P2P) video streaming access control, and designs a novel decision-making mechanism using formal fuzzy rules and reasoning mechanisms to adjust P2P video streaming resource allocation following individual users' preferences.

1 Introduction

With a fast development of P2P technology, various applications based on P2P have enriched people's daily lives. P2P-VoD such as PPLive, PPStream and TVAnt are among the most popular applications. The system described in [1] has been proved that it can offer users a fast, convenient and on-demand video streaming experience. However, as illustrated in [2], users may suffer attacks from other peers when they are enjoying videos. Thus, the reliability of video streaming sources becomes one of the biggest issues in P2P-VoD. Like other P2P applications, how to allocate video streaming resources is another issue. In this paper, the trustworthiness of people in online social network is applied into a P2P-VoD system to ensure the reliability of video streaming sources, since the trustworthiness between people in online social networks are widely used and adopted by people. As described in [3], the relationships between users in online social networks are more complex than a simple indication of two people's mutual acquaintance, and this complexity may cause confusion when users have to determine trust degrees between each other. Users also may remain uncertain when allocating resources to

Zhengping Wu · Hao Wu

Dept. of Computer Science and Engineering, University of Bridgeport

e-mail: zhengpiw@bridgeport.edu, wuhao@bridgeport.edu

peer users, since the capacity of a transmission channel, the reliability of that channel, and local or remote resources may vary. To manage a collection of uncertain factors in the decision-making process for access control and resource allocation in P2P video streaming, we provide an adaptive mechanism using users' own preferences represented in decision-making policies. We introduce "fuzziness" into policy representation and enforcement in this mechanism. Applying fuzzy logic into policy-based access control and resource allocation for P2P video streaming can help users handle multiple factors in decision-making activities with a certain level of uncertainty. We propose and implement a P2P video streaming system over online social networks using policy-based access control and resource allocation, which combines various types of control information following users own preferences based on fuzzy enforcement of policies containing uncertain factors.

2 Access Control and Resource Allocation

Since we simulate the trust in P2P video streaming by using trust relationships from online social networks, we need to understand trust in online social networks first. Some online social networks allow users assigning trust levels for friends and other users [5]. Golbeck and Hendler propose a binary method to calculate the trust between users in online social networks [3]. Other methods such as the ones described in [6] also provide algorithmic supports for calculating trust between users. But in most online social networks, trust between users is simply calculated from the level of friend-of-a-friend (FOAF) relationships. In this paper, we also calculate trust between people from the level of FOAF relationships.

Following the categorization described by Beth et al. [4], we categorize trust into two classes - direct trust and indirect trust. A trust relationship formed from direct experience or negotiations can be characterized as direct trust; a trust relationship or a potential one built from recommendations by a trusted third party or a chain of trusted parties, which create a trust path, is called indirect trust. Based on FOAF relationships, trust between people can also be classified into direct and indirect trust. As discussed before, indirect trust relationships may form from different paths, and the degree of each path may vary. So users cannot always clearly tell the trustworthiness between people in online social networks, because everyone has his or her own opinion on trust, and he or she may remain uncertain when defining trust between users in online social networks. We introduce fuzzy logic into the definition of trustworthiness into our framework, and provide a binary access control decision for P2P video streaming based on the degree of trustworthiness between peer users.

Users may feel the confusion when allocating resources for P2P video streaming since multiple factors need to be considered. For instance, the actual bandwidth for P2P video streaming between peer users may be impacted by the communication channel's capacity, the communication channel's reliability, and access point control. The channel capacity determines the information transmission speed; the channel reliability determines the integrity and safety of transferred information; access point control determines the trustworthiness of peer users and their access privileges. So we also use fuzzy logic in our resource

allocation process, which uses non-quantitative information. After the access control process, these peers who are allowed to share a video stream will go through the resource allocation process, which combines the trustworthiness of peers and current resource usage status, and then automatically allocate resources or adjust resource usage following individual peer users' preferences.

3 Fuzzy Representation and Enforcement

3.1 Fuzzy Model of Uncertainty

The trust relationships in P2P video streaming are hard to assess due to the uncertainty involved. If a trust relationship relies upon a subjective judgment based on indirect information, it will be very uncertain and any access operations related to that trust relationship may cause unexpected results.

Fuzzy logic is a suitable way to represent uncertainty, especially when they need to be handled quantitatively. By introducing fuzzy logic into the research of trustworthiness, we try to solve the issues associated with uncertainty management in P2P video streaming applications. First, we need to identify the subjects of these issues. These subjects are either the sources of trust-related information needed in the management of trust or the entities with which trust relationships are built. The set of these subjects is represented as M in this paper. Then we need to define a general fuzzy set in the management of trust.

Definition 1. Fuzzy set for management of trust

For every element m in the set of subjects M , there is a mapping $m \rightarrow f(m)$, in which $f(m) \in [0,1]$. The set $\Delta = \{(m, f(m))\}$ for $\forall m \in M$ is defined as a fuzzy set for management of trust. $f(m)$ is defined as the membership function for every m in Δ .

All the fuzzy sets on M are represented as $F(M)$. Then we can use a group of fuzzy sets from $F(M)$ to group all the elements of M into several sets with different levels of uncertainty.

In real life, the level of uncertainty cannot be limited to only one set, and the degrees to these sets are not simply 'total' or 'none'; additionally, it is sometimes difficult to determine which set or sets should be used for certain types of uncertainty. So a vector consisting of the degrees of belongingness to each uncertainty set $D = \{d_1, d_2, d_3\}$ is more appropriate for describing the actual trustworthiness-based judgment from daily life, in which d_i ($i = 1, 2, 3$) is the degree of belongingness to set Z_i ($i = 1, 2, 3$). Meanwhile, there are several ways to determine or calculate individual degrees d_i . One way is direct judgment that determines the degree from direct experience or evaluation. Another one is indirect inference that determines the degree via an analysis of an indirect source such as reputation or recommendation. The first one is relatively subjective while the evaluation method may be very objective; and the second one is relatively objective while the source of information is indirect. Other ways to determine the degrees also exist, which will not be discussed in this paper.

3.2 Fuzzy Representation of Uncertainty

To reason among the degrees of uncertainty in trust management for further inference or decision-making, we need to represent uncertainty formally. Direct trust is formally described as $x \xrightarrow{D} y [Z]$, which means entity x is willing to rely upon entity y to degree D for the categorized uncertainty Z . D is a vector with corresponding degrees of belongingness for each set in categorization Z . Direct trust is from direct experience with the trustworthiness of the other entity or from a judgment with subjective/objective evaluation. Indirect trust is described as $x \xrightarrow{P} y [Z]$,

which means entity x is willing to rely upon y to degree D following P 's recommendation for the categorized uncertainty Z . P is one or more entities constructing a path that gives a recommendation to entity x for entity y . D is a vector with corresponding degrees of belongingness for each set in categorization Z . Indirect trust is derived from the recommendation passed through one or more intermediate entities. There are also two types of recommendations. One type is that the recommender had direct experience with the recommended entity so that the P has only one entity; the other is that the final recommender formed the recommendation from further recommendations of other recommenders so that the P has more than one entity constructing a chained recommending path or a compound recommending graph. But from the recommendee's (entity x 's) point of view, there is no big significance related to with the number of entities forming the recommending path; the recommendee (entity y) only cares about the final recommender's capability to make accurate recommendation based on its own experience and trustworthiness.

The use of fuzzy logic to describe uncertain rules in trust management can use antecedent terms with the below form:

“If the confidence (of some event) is high, then a certain action is performed.”

We apply this general form to describe fuzzy rules in a trustworthiness-based decision-making system for video streaming to express real life uncertainty in trust management and decision-making with human linguistics. Here different formats of the probability function W introduce different types of rules. If W is a threshold function, the rule becomes a threshold decision rule; if W has a fuzzy definition, the rule is a fuzzy rule.

3.3 Fuzzy Enforcement

To adapt to different sources of uncertainty in trust management, a parameterized general intersection operator and union operator are needed. They are also called T-norm and S-norm. With different values of the parameter, these operators can maximize the expressiveness and flexibility of the system to capture people's intentions toward these uncertainties. As described in [8], here we use a parameter $\alpha \in [0,1]$ to adjust T-norm and S-norm. Then we define two calculators on vectors of fuzzy values - connection calculator and union calculator, which use T-norm and S-norm respectively. After we define the above calculators, we can perform formal

analysis on fuzzy sets and fuzzy rules used for uncertainty expressions. Here we define two sets of derivation rules (deduction rules and consensus rules) to handle different types of uncertainty. Below are the formal descriptions of deduction rules.

Definition 2. Deduction rules

$$\begin{aligned}
 x \xrightarrow{D} y[Z] \wedge y \xrightarrow{D'} z[Z] &\Rightarrow x \xrightarrow{D'} z[Z] \wedge (P' = \{y\}) \wedge (D' = D \otimes D) \\
 x \xrightarrow{D} y[Z] \wedge y \xrightarrow{D'} z[Z] &\Rightarrow x \xrightarrow{D'} z[Z] \wedge (P' = \{y, P'\}) \wedge (D' = D \otimes D) \\
 x \xrightarrow{D} y[Z] \wedge y \xrightarrow{D'} z[Z] &\Rightarrow x \xrightarrow{D'} z[Z] \wedge (P' = \{P, P'\}) \wedge (D' = D \otimes D)
 \end{aligned}$$

Deduction rules are used for a recommendation's connection to construct a whole recommendation chain that allows the trustworthiness to be transferred from one end to the other end. For the trust relationships from the same categorization, deduction rules can form a new connection using the trust relationship between the recommender and the recommendee and embed the content of that recommendation into the new connection. Below are the formal descriptions of consensus rules.

Definition 3. Consensus rules

$$\begin{aligned}
 x \xrightarrow{D_1} y[Z] \wedge x \xrightarrow{D_2} y[Z] \wedge \dots \wedge x \xrightarrow{D_n} y[Z] &\Rightarrow x \xrightarrow{D'} y[Z] \wedge (D' = D_1 \oplus D_2 \oplus \dots \oplus D_n) \\
 x \xrightarrow{D_1} y[Z] \wedge x \xrightarrow{D_2} y[Z] \wedge \dots \wedge x \xrightarrow{D_n} y[Z] &\Rightarrow x \xrightarrow{D'} y[Z] \wedge (P' = \{P_m \mid |P_m| = \min\{P_i \mid (i = 1..n)\}\}) \\
 &\wedge (D' = D_1 \oplus D_2 \oplus \dots \oplus D_n)
 \end{aligned}$$

Consensus rules are used for combining of multiple recommendations for the same kind of categorization. When two or more recommendation paths appear simultaneously, consensus rules can synthesize the opinions to form a comprehensive recommendation. The shortest recommending path is the easiest path to verify that indirect information, even if the value of the trust degree vector is not as high as others. We use this path as the recommending path for verification of that recommendation. But more likely we will only use the unified trust degree vector alone after the composition.

4 System Architecture and Implementation

With the help of the fuzzy operations and rules defined above, we can form a formal decision-making process to handle uncertainty in the management of trustworthiness for access control and resource allocation. Users need to define the categorization of uncertainty. Then the decision-making process uses fuzzy operations to combine uncertain information from different sources. After defuzzification, users need to judge whether the final decision is consistent with their own rules. If not, the parameters of the fuzzy operations need to be adjusted. The entire decision-making process is illustrated in figure 1 (mainly on the left part).

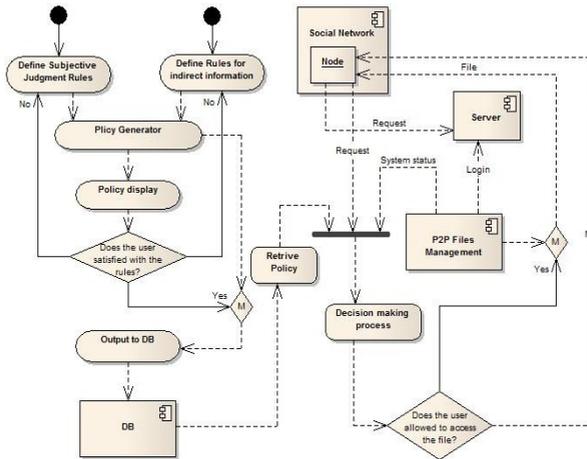


Fig. 1 System Architecture

Following the decision-making process, we illustrate some practical fuzzy policies, the user interface to input fuzzy policies, and the enforcement mechanism to enforce these policies for P2P video streaming. The prototype system is implemented as an “application” in Facebook. We provide an interface for users to share videos with their Facebook friends and other peer users who are using the same application. Users can search and watch their friends’ shared videos. We design and implement a user interface to assist users to input these policies consistent with the accurate rules or preferences in their minds. Users can also modify membership functions to align with their own intentions rather than the default functions, and combine these membership functions using “and” or “or” operator between any two fuzzy terms for different factors. After a user defining all the policies and membership functions, the system uses a policy generator to translate these fuzzy policies into a formal format, and store them in a policy database. Once the users start watching videos, the system will adjust the connection status automatically using these policies. Users can also train the system by adjusting certain parameters in the decision-making process to refine system’s accuracy.

5 A Case Study and Experiments

To examine the performance and usability of the system, we set up an experimental environment including 30 users and implement the P2P video streaming control framework in a video sharing system.

First, we tested our system on two different policy sets: first one is following the common sense that upload speeds vary as the level of trustworthiness changes; the second one prefers the most trustworthy peer users. Both policy sets prevent

untrusted peers to connect to the system. We establish a peer user, and allow this user sharing videos with others. Then four other peer users with different trust levels are connected to this user to watch videos. As figure 2 (two upper charts) shows, only three of them can access the system, the one not trusted by the first established peer is blocked to enter the system. When the number of connected peers increases, the download speed for the peer user with a relative low trust degree will drop dramatically, if the system adopts the first policy set. However when the system adopts the second policy set, only trusted peer users can get a relatively good download speed. The speeds of other two peer users drop to almost the same level. The result shows the system’s flexibility, which reflects users’ preferences represented in different policy sets, and the system’s accuracy to capture and enforce users’ preferences.

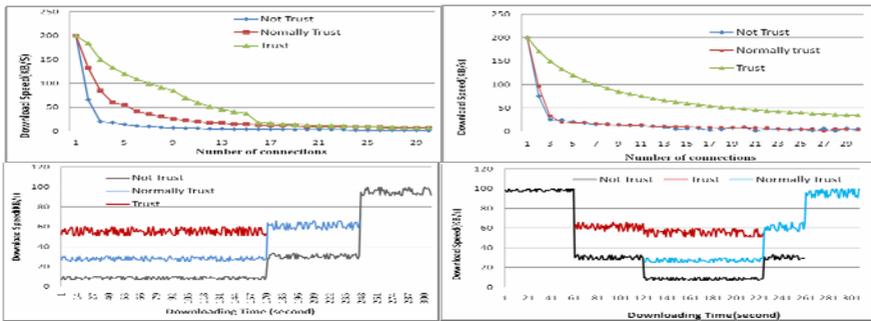


Fig. 2 Experiment Results

Following the first experiment on different policy sets, we perform the second experiment on watching video from different peer users with different trust levels to examine the flexibility and adaptability of the system. This experiment is based on the first policy set. As illustrated in figure 2 (two lower charts), when these three peer users start watching video at the same time, the trusted peer user can get the highest priority, and the normally trusted peer user gets higher priority than the peer with lower trust degree. After the trusted peer user finishes watching and releases the occupied resources, the other two peer users are allocated more bandwidth, and their download speeds increase with the same amounts. Then these three peer users start watching videos from different time. As figure 2 (two lower charts) shows, the bandwidth of one with lower trust level dropped to the very bottom when other two are connected to the system no matter who connects first. These results show that the system can automatically allocate and adjust download speed to different peer users according to the policy (preferences) defined by the established user, which truly provides a flexible control of P2P video streaming.

6 Conclusion

In this paper, a framework of adaptive control based on fuzzy logic is proposed and implemented to handle uncertainty and fuzziness in the decision-making process of access control and resource allocation for P2P video streaming. This paper uses relationships from online social networks to simulate trust in P2P video streaming, which addresses the issue how people can trust their peer users. This paper introduces membership functions from fuzzy logic into policy definition to describe uncertainty and fuzziness in trustworthiness, and defines a trust degree vector to evaluate different levels of trustworthiness. In addition, the derivation rules proposed in this paper incorporate a parameter to allow users to adjust the membership function through a feedback mechanism in order to make the system adapt to users' changing preferences and actual application environment requirements. The model proposed in this paper can be used in evaluation, analysis, and derivation of policies in preference management directly. The experiments in the previous section show the accuracy, flexibility, usability and adaptability of the system, and also prove that using trust relationships from social networks can accurately model the trustworthiness between peers in P2P video streaming. This work can also provide effective support for policy-based management of uncertainty and decision-making in preference management for other applications with uncertain and fuzzy factors.

References

- [1] Huang, Y., Fu, T.Z.J., Chiu, D.-M., Lui, J.C.S., Huang, C.: Challenges, design and analysis of a large-scale p2p-vod system. In: Proceedings of the ACM SIGCOMM 2008 conference on Data communication, pp. 375–388 (2008)
- [2] Jia, J., Chen, C.: A mesh check scheme against P2P live streaming attacks. In: Proceedings of the 5th International ICST Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (2008)
- [3] Golbeck, J., Hendler, J.: Inferring Binary Trust Relationships in Web-Based Social Networks. *ACM Transactions on Internet Technology* 6(4), 497–529 (2006)
- [4] Beth, T., Borcherding, M., Klein, B.: Valuation of Trust in Open Networks. In: Proceedings of the 1994 European Symposium on Research in Security, pp. 3–18 (1994)
- [5] Sabater-Mir, J.: Towards the next generation of computational trust and reputation models. In: Torra, V., Narukawa, Y., Valls, A., Domingo-Ferrer, J. (eds.) *MDAI 2006*. LNCS (LNAI), vol. 3885, pp. 19–21. Springer, Heidelberg (2006)
- [6] Parreira, J.X., Donato, D., Castillo, C., Weikum, G.: Computing trusted authority scores in peer-to-peer web search networks. In: Proceedings of the 3rd International Workshop on Adversarial Information Retrieval on the Web, pp. 73–80 (2007)
- [7] Weber, S.: A general concept of fuzzy connectives, negations, and implications based on t-norms. *Fuzzy Sets System* (11), 115–134 (1983)
- [8] Wu, Z., Wu, H.: Uncertainty Management in Personalized Recommendation for E-Commerce Applications. In: Proceeding of 21st International Conference on Tools with Artificial Intelligence (2009)

Bio-inspired Optimization Methods of Fuzzy Logic Controllers Applied to Linear Plants

Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, Antonio Rodriguez

Abstract. We use an optimization method to find the parameters of the membership functions of a fuzzy logic controller (FLC) in order to minimize the steady state error for linear systems. The genetic algorithm (GA) and particle swarm optimization (PSO), which are optimization methods, were used to find the optimal FLC. The obtained FLC achieves regulation of the output and stability of the closed-loop system. Simulation results, with the optimal FLC and implemented in Simulink, show the feasibility of the proposed approach.

1 Introduction

Some optimization methods are based on populations of solutions. Unlike the classic methods of improvement for trajectory tracking, in this case in each iteration of the algorithm, it does not have a unique solution but a set of solutions. These methods are based on generating, selecting, combining and replacing a set of solutions. Since they maintain and they manipulate a set instead of a unique solution throughout the entire search process, they used more computer time than other metaheuristic methods. This fact can be aggravated because the “convergence” of the

Ricardo Martinez · Antonio Rodriguez
Universidad Autónoma de Baja California Tijuana, México
e-mail: mc.ricardo.martinez@hotmail.com, ardiaz@uabc.mx

Oscar Castillo
Tijuana Institute of Technology, Tijuana México
e-mail: ocastillo@hafsamx.org

Luis T. Aguilar
Instituto Politécnico Nacional, Centro de Investigación y Desarrollo de Tecnología Digital,
Ave. Del Parque 1310, Mesa de Otay, Tijuana B.C. 22510,
Fax: +52(664)6231388
e-mail: luis.aguilar@ieee.org

population requires of a great number of iterations. For this reason a concerted effort has been dedicated to obtaining methods that are more aggressive and manage to obtain solutions of quality in a nearer horizon. This study is about PSO [11] and GA [11] applied to the optimization of fuzzy controllers for linear plants. To this end, we consider a type-2 fuzzy logic system to obtain the optimal controller. Previously, we optimize a type-2 fuzzy logic controller for an autonomous mobile robot for trajectory tracking, where the genetic algorithms were used to find the optimal controller obtaining good results under several kinds of perturbations [6]. For this study, we use transfer functions to test the optimal type-2 fuzzy logic controllers. We consider two transfer functions one of them it's more complicate to get the control. Each of the optimization methods, the GA and PSO are used to find the parameters of the membership functions. We made a comparison of the FLC obtained with genetic algorithm and particle swarm optimization for the plant control.

This paper is organized as follows: Section 2 presents the theoretical basis and problem statement. Section 3 introduces the controller design where a genetic algorithm is used to select the parameters. Robustness properties of the closed-loop system are achieved with a type-2 fuzzy logic control system using a Takagi-Sugeno model where the error and the change of error, are considered the linguistic variables. Section 4 provides a simulation study of the plant using the controller described in Section 3. Finally, Section 5 presents the conclusion.

2 Theoretical Basis and Problem Statement

Evolutionary Methods Applied to Fuzzy Systems. Fuzzy systems have been successfully applied to problems in classification [4], modeling, control [6], and in a considerable number of applications. In most cases, the key for success was the ability of fuzzy systems to incorporate human expert knowledge. In the 1990s, despite the previous successful history, the lack of learning capabilities characterizing most of the works in the field generated a certain interest for the study of fuzzy systems with added learning capabilities. A GFS is basically a fuzzy system augmented by a learning process based on a genetic algorithm (GA) [1], [2], [3], [5].

Particle Swarm Optimization (PSO). PSO is a population based stochastic optimization technique developed by Eberhart and Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling [7]. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA) [8], [9] [10].

Problem Statement. We used two linear plants (transfer functions) with a different level of complexity for test the optimization methods.

- Plant 1 (that is stable) is given by the following second order transfer function:

$$g(s) = \frac{w_n^2}{s^2 + 2\epsilon w_n s + w_n^2} \quad \epsilon = 0.5, \quad w_n = 2 \quad (1)$$

where w_n is the natural frequency and ϵ is the coefficient damping.

- Plant 2 (that is unstable) is given by the following transfer function:

$$g(s) = \frac{1}{s^2 + 4} \tag{2}$$

3 Fuzzy Logic Control Design

In this section we design a fuzzy logic controller (FLC) where the optimal controller was found with the first evolutionary method, which in this case is the genetic algorithm.

For the FLC a Takagi-Sugeno type of fuzzy systems is used with two inputs a) error, and b) error change, with three membership functions each input, “Negative, Zero and Positive” (Gaussian and triangular), one output which are constant values, and nine fuzzy rules (IF – THEN). Fig 1 shows the FLC membership functions for the plant control. Once we obtained the FLC design, we used an optimization method to find the optimal Controller.

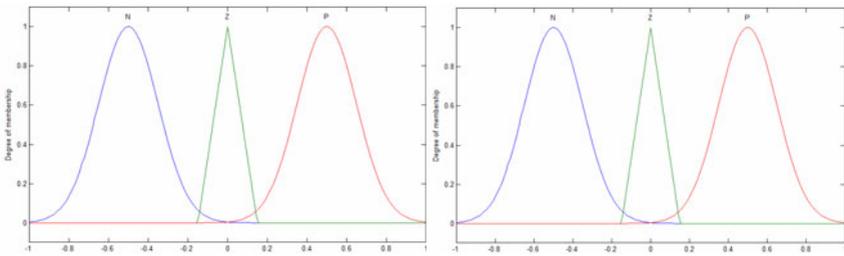


Fig. 1 a) input 1 “error”, b) input 2 “error change”

Genetic Algorithm: The genetic algorithm chromosome has 17 genes of real values and they represent the two inputs, error and error change and one output constant values. Fig 2 shows the chromosome representation for the FLC.

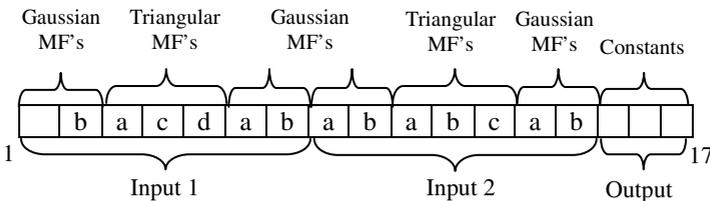


Fig. 2 Chromosome representation for the fuzzy logic controller

PSO: For this optimization we used particle swarm to find the optimal parameters of the membership functions of the FLC controller changing the cognitive and social variables of the PSO.

Table 1 shows the parameters of the membership functions, the minimal and the maximum values in the search range for the genetic algorithm and PSO methods to find the best fuzzy controller system.

Table 1 Parameters of the membership functions

Plant 1				Plant 2			
MF Type	Point	Minimum Value	Maximum Value	MF Type	Point	Minimum Value	Maximum Value
Gaussian	a	0.3	0.6	Gaussian	a	1.8	2.8
	b	-1	-1		b	-5	-5
Triangular	a	-0.3	-0.8	Triangular	a	-0.5	-4
	b	0	0		b	0	0
	c	0.3	0.8		c	0.5	4
Gaussian	a	0.3	0.6	Gaussian	a	1.8	2.8
	b	1	1		b	5	5

4 Simulations Results

In this section, we evaluate, through computer simulations performed in MATLAB® and SIMULINK®, the designed FLC for the two plants using Genetic Algorithms and PSO optimization methods.

4.1 Genetic Algorithm Results

Table 2 presents the main results of the FLC obtained with the genetic algorithms showing in the third row the best result.

Table 2 Results of the FLC obtained by genetic algorithms

No.	Indiv.	Gen.	% Remp.	Cross.	Mut.	GA Time	Average error
1	90	35	0.7	0.6	0.3	00:18:06	0.050870
2	150	80	0.7	0.5	0.2	01:19:13	0.044310
3	80	50	0.7	0.5	0.2	00:23:01	0.071366
4	45	60	0.7	0.6	0.3	00:16:03	0.068477
5	75	50	0.7	0.6	0.1	00:21:37	0.068158

Fig 3 shows the evolution of the genetic algorithm giving the best FLC for controlling the plant 1.

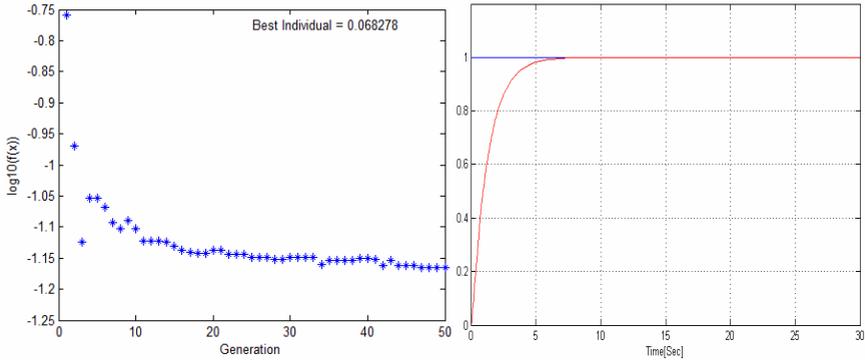


Fig. 3 Evolution of the GA for the FLC optimization and simulation result

Table 3 presents the main results of the obtained with the genetic algorithms of Plant 2.

Table 3 Results of the FLC obtained by genetic algorithms

No.	Ind.	Gen.	% Remp.	Cross	Mut.	GA Time	Average error
1	200	90	0.5	0.4	0.1	01:33:19	0.065268
2	120	85	0.5	0.4	0.1	00:52:13	0.070636
3	90	35	0.5	0.5	0.2	00:16:26	0.074058
4	65	35	0.5	0.4	0.2	00:11:53	0.076711
5	55	45	0.5	0.6	0.1	00:39:01	0.077597

Fig 4 shows the evolution of the genetic algorithm and simulation result of the optimized FLC of Plant 2.

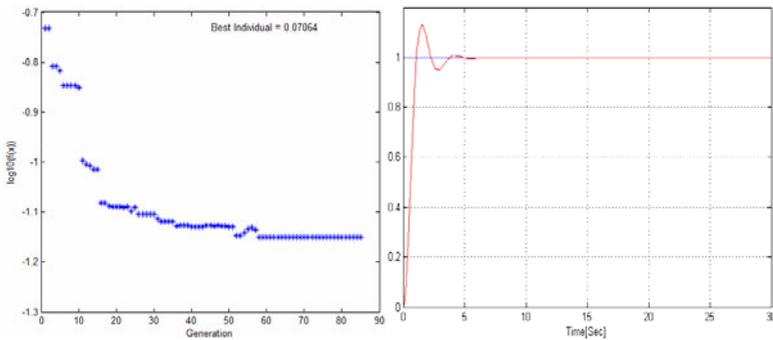


Fig. 4 Evolution of the GA of FLC optimization and Simulation result

4.2 PSO Results

Table 4 presents the main results of the FLC obtained with PSO method showing the best result in the fourth row.

Table 4 Results of the FLC obtained by PSO

No.	Population Swarm	Max Iterations	C1	C2	Inertia	time exec	Average error
1	200	70	1	1	1	0:05:27	0.06321
2	200	70	0.5	0.5	1	0:07:35	0.05276
3	200	70	0.25	0.25	1	0:15:35	0.05576
4	200	70	0.15	0.15	1	0:19:15	0.09717
5	200	70	0.05	0.05	1	0:26:10	0.09400
6	200	70	0.005	0.005	1	0:26:56	0.11377

Fig 5 shows the behavior particles of PSO giving the best FLC for controlling the plant 1.

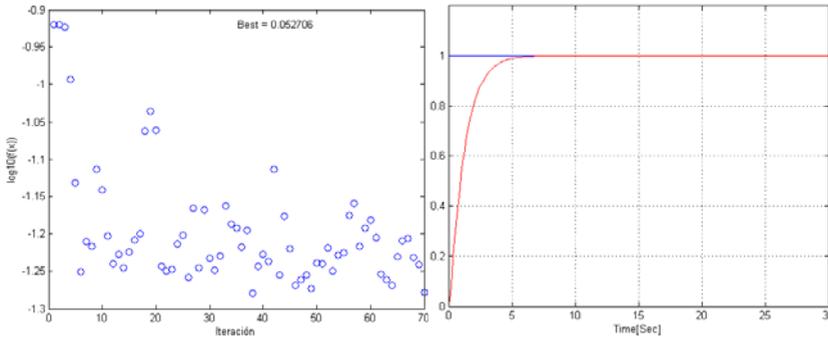


Fig. 5 Behavior of the particles of PSO for the FLC optimization and the simulation result

Table 5 presents the main results of the obtained with the PSO method of Plant 2 showing the best result in the fifth row.

Table 5 Results of the FLC obtained by PSO

No.	Population Swarm	Max Iterations	C1	C2	Inertia	time exec	Average error
1	200	70	1	1	1	0:04:59	0.14286
2	200	70	0.5	0.5	1	0:07:11	0.12527
3	200	70	0.25	0.25	1	0:10:21	0.14508
4	200	70	0.15	0.15	1	0:16:25	0.15090
5	200	70	0.05	0.05	1	0:17:56	0.13431
6	200	70	0.005	0.005	1	0:21:18	0.14513

Fig 6 shows the behavior particles of PSO and simulation result of the optimized FLC of Plant 2.

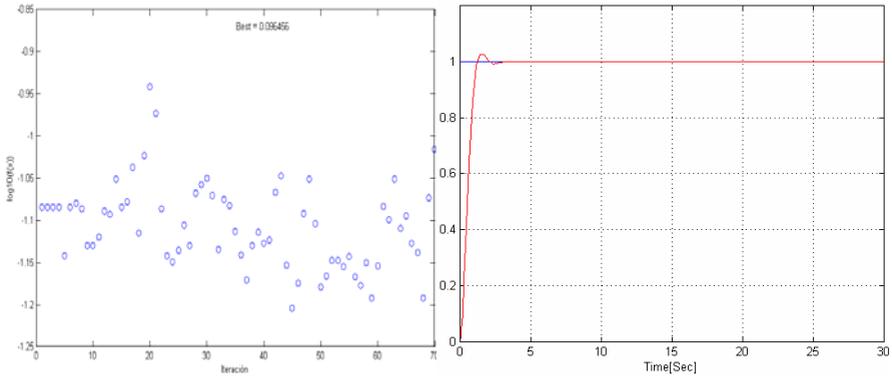


Fig. 6 Behavior of PSO particles of FLC optimization and the Simulation result

5 Conclusions

We described the use of optimization methods for the simulation design of optimized FLC's. In particular we presented results of a genetic algorithm and PSO in Type-1 FLC optimization for linear plants. The simulation result shows that the closed-loop system with the FLC obtained by GA and PSO is stabilized in less than 10 sec. On the other hand, the FLC's obtained by PSO are better than the FLC's obtained by GA because PSO had less time consuming in the process and a minor overshoot in the second Plant, the plots of the results shows this difference.

We have achieved satisfactory results with genetic algorithms and PSO; the next step is to solve the problem using Type-2 FLC in perturbed environment and multiple objective optimizations to obtain better results. Moreover, we will extend the results for nonlinear systems that autonomous mobile robot.

References

1. Alcalá, R., Alcalá-Fdez, J., Herrera, F.: A proposal for the Genetic Lateral Tuning of Linguistic Fuzzy Systems and its Interaction with Rule Selection. *IEEE Transactions on Fuzzy Systems* 15(4), 616–635 (2007)
2. Bentalba, S., El Hajjaji, A., Rachid, A.: Fuzzy Control of a Mobile Robot: A New Approach. In: *Proc. IEEE Int. Conf. On Control Applications*, Hartford, CT, October 1997, pp. 69–72 (1997)
3. Casillas, J., Cordon, O., del Jesús, M.J., Herrera, F.: Genetic Tuning of Fuzzy Rule Deep Structures Preserving Interpretability and its Interaction with Fuzzy Rule Set Reduction. *IEEE Transaction on Fuzzy Systems* 13(1), 13–29 (2005)
4. DeJong, K.: Learning with genetic algorithms: an overview. *Mach. Learning* 3(3), 121–138 (1988)

5. Driankov, D., Hellendoorn, H., Reinfrank, M.: *An Introduction to Fuzzy Control*. Springer, Berlin (1993)
6. Fukao, T., Nakagawa, H., Adachi, N.: Adaptive Tracking Control of a Non-Holonomic Mobile Robot. *IEEE Trans. On Robotics and Automation* 16(5), 609–615 (2000)
7. Fogel, D.B.: An introduction to simulated evolutionary optimization. *IEEE transactions on neural networks* 5(1), 3–14 (1994)
8. Angeline, P.J.: Evolutionary Optimization Versus Particle Swarm Optimization: Philosophy and Performance Differences. In: Porto, V.W., Waagen, D. (eds.) *EP 1998*. LNCS, vol. 1447, pp. 601–610. Springer, Heidelberg (1998)
9. Kennedy, J., Mendes: The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation* 6(1), 58–73 (2002)
10. Kennedy, J., Mendes, R.: Population structure and particle swarm performance. In: *Proceeding of IEEE conference on Evolutionary Computation*, pp. 1671–1676 (2002)
11. Gomez Blas, N., De Mingo, L.F., Castellanos Peñuela, J.: *Bio-Inspired Optimization Strategies: A Survey*, <http://csit.am/2009/proceedings/4AIMSS/2.pdf>

Part VII

**Special Session Traffic and
Transportation Systems**

An Adaptive Neuro-Fuzzy Inference System for Simulation of Pedestrians Behaviour at Unsignalized Roadway Crossings

Michele Ottomanelli, Leonardo Caggiani, Giuseppe Iannucci,
and Domenico Sassanelli

Abstract. “Gap acceptance” behaviour oversees pedestrians crossing manoeuvre at unsignalized road crossings. From a scientific point of view, the study of pedestrians behaviour has a particular interest, since the underlying factors of behavioural interaction between pedestrians and motor vehicles drivers have a strong non-deterministic component, which makes their simulation very complex. In this paper a Fuzzy logic model for representation and simulation of pedestrian behaviour in such a manoeuvre is proposed. The calibration of Fuzzy model membership functions is executed through an Adaptive Neural Network which considers a sample of “gap acceptance” decisions collected on field. The analysis method is at first theoretically defined and then applied to a real pedestrian crossing.

1 Introduction

The issue of accidents on pedestrian crossings is highly emotional and raises a very strong interest within the public opinion and the media, mainly because the victims are often children and elderly people. Furthermore, everybody travelling is sooner or later a pedestrian himself before and after using any kind of vehicle. In 2005, pedestrian fatalities [1] (in and outside pedestrian crossings) per 1 million population in Europe has rates between 15.7% and 4.6%. Of these, pedestrians

Michele Ottomanelli
Dept. of Environmental Engineering and Sustainable Development
Technical University of Bari - Viale De Gasperi - 74100 Taranto, Italy
e-mail: m.ottomanelli@poliba.it

Leonardo Caggiani · Giuseppe Iannucci · Domenico Sassanelli
Dept. of Roads and Transportation
Technical University of Bari - Via Orabona, 4 - 70125 Bari, Italy
e-mail: l.caggiani@poliba.it, g.iannucci@poliba.it,
d.sassanelli@poliba.it

fatalities occurred at pedestrian crossings represent a ratio between 37% (Norway) and 6 % (Netherlands).

In the simulation field, the decision-making processes that oversee the implementation of pedestrians crossing manoeuvre, as well as the assessment of opportunity to stop the vehicle or to slow down the march in the driver case, have particular interest. Usually in the field of transport, human behaviour (both in the case of pedestrians and drivers) at a crossing (pedestrian or road intersection) has been usually described through the so-called theory of “gap acceptance”, based on a probabilistic/deterministic approach.

In this paper a different method of reproduction of pedestrian behaviour is proposed. The method is based on the assumption that when a pedestrian crosses the road, he evaluates a series of qualitative parameters that describe the closest vehicle motion rather than a numerical estimation of them. These subjective evaluations are inevitably affected by a considerable degree of uncertainty, and they cannot be merely expressed with probabilistic models, as they depend on the intrinsic vagueness concerning pedestrians assessment.

2 Statement of the Problem – Gap Acceptance Theory

In literature, gap acceptance problem has been largely analyzed for signalized or unsignalized intersections, but mainly from the driver’s point of view. A complete description of principal aspects of driver behaviour is presented in [2]. Since a pedestrian behaviour is similar to a driver’s one, pedestrian crossing behaviour can be considered largely governed by gap acceptance theory too [3, 4, 5]. This theory states that each pedestrian or a group of pedestrians [6, 7] has a critical gap. Arrived at the curb, the pedestrian checks if the current traffic gap is greater than the critical gap and decides whether to accept the traffic gap. If the current one is rejected, the next one is considered. This process continues until the pedestrian accepts a traffic gap or gives up. Generally the critical gap numerical value is not constant among pedestrians and it often varies also for the same individual, due to subjectivity and to lack of pedestrians consistency. [2]. The critical gap consists of two parts. One is the required crossing time and the other is a safety margin, often defined aggressiveness factor. The safety margin is the difference between the time a pedestrian crosses the traffic and the time the next vehicle arrives at the crossing point. The crossing time is what it takes a pedestrian to cross a particular road. This theory, thus, indicates that pedestrians crossing behaviour is governed largely by three components: the supply of gaps, the crossing time, and the safety margin. According to this theory, a pedestrian evaluates his Estimated Crossing Time (ECT), that is his critical gap, compares it with the Available Time between two Vehicles arriving at the crossing (TVA), and decides if to cross [8]. This can be summarised in the following rule: “*if TVA is greater than ECT then CROSS, Otherwise, WAIT*”, where ECT can be expressed as the sum of time needed to cross, depending on the pedestrian walking speed (V), and the crossing Aggressiveness Factor (AF). It turns out:

$$ECT = \frac{L}{V} + AF \quad (1)$$

where L is the crossing length.

Due to the variability of the critical gap (ECT) among pedestrians some authors suggest to estimate the cumulative distribution function of ECT. Logit model (Gumbel distribution) [2, 9], probit model [2, 10], maximum likelihood calibration [2, 11, 12] or regression models [13] have been proposed. Distributions trend and their mean values depend on different geometrical and subjective factors. Only few authors consider fuzzy logic to simulate driver's gap perception uncertainty [14, 15].

3 Proposed Model – Fuzzy Logic Approach

Human decisions are not based on numerical parameters judgments, on the contrary they are based on qualitative opinions. So pedestrians do not really compare the numerical value ECT with TVA, but they perform qualitative evaluations, that can be more conveniently expressed in the linguistic form with expressions like “a lot of time to cross”, “not much time to cross” etc. These adjectives (a lot of, not much, low, high...) can be represented with fuzzy sets. In order to simulate the fuzzy judgments of the pedestrian it is necessary to establish a Fuzzy Inference System (FIS). The Fuzzy Logic pedestrian behaviour simulation model proposed, assumes that the pedestrian decision is based on qualitative assessment of motion parameters of the vehicle immediately closer to the crossing. These parameters are evaluated when the pedestrian decides to begin the crossing manoeuvre, and are the *vehicle's speed*, the *vehicle's distance* from the crossing and the *interval* between vehicle arrival and pedestrian arrival at the crossing (“lag”) or, if the first interval is rejected, interval between two consecutive vehicles (“gap”).

Each combination of these parameters implies the development of an “ad hoc” FIS composed by a particular combination of input parameters and a set of rules. In this paper two fuzzy inference systems for simulation of pedestrians decisions behaviour are specified.

3.1 FIS Model A

In the first FIS proposed (Model A) we consider the case in which a pedestrian evaluates only the *gap* time or *lag* time. Five membership functions that bind the possible judgments made by a pedestrian on lag or gap values have been defined. The shapes of membership functions chosen are triangular and trapezoidal and are designed, as first attempt, on the basis of “expert” assessments. Of course, fuzzy values of intervals must be referred to a particular pedestrian crossing. For each linguistic variable the more the degree of membership is equal to one the more the corresponding *gap* or *lag* value belongs to the respective linguistic variable. We get two singleton outputs (decision) from fuzzy model (*wait* is equal to zero and *cross* is equal to one). Five logical rules have been applied such as “If *Gap* or *Lag* is low then *decision* is *wait*”.

3.2 FIS Model B

In the second FIS proposed (Model B) we assume that pedestrians consider both lag/gap and speed of vehicle approaching (in literature only few authors consider more than one parameter at the same time [15]). Even in this case on the basis of “expert” assessments, we fixed five membership functions respectively for gap/lag and for speed. Also in this case *wait* is equal to zero and *cross* is equal to one. Twenty-five logical rules have been applied such as “If *Gap or Lag* is *very low* and *Speed* is *very low* then *decision* is *cross*”.

3.3 Model Calibration – ANFIS

After setting the architecture of the models (input and output parameters, membership functions shapes and logical rules), it is necessary to calibrate membership functions parameters (position of triangles and trapezoids vertexes) because they have been chosen on the basis of few “expert” pedestrians opinions and could differ from those regarding a larger pedestrians population. To operate this calibration the ANFIS (Adaptive Neuro-Fuzzy Inference System) methodology was considered, adopting a neural network to calibrate FIS membership functions.

4 Case Study, Model Specification and Application

A pedestrian crossing located in the municipality of Bari (Italy), placed at a pedestrian entrance of the Technical University of Bari was considered as a case study. The crossing was chosen because of high pedestrian flows, combined with high vehicular flows. The crossing is 4.00 meters wide and has a length of 10.30 m. The road is travelled by vehicles in one way.

In the proposed models, the ANFIS algorithm has been specified with one input (gap/lag) and one output (decision: cross = 1; wait = 0) for Model A and two input (gap/lag and speed of approaching vehicle) and one output (decision: cross = 1; wait = 0) for Model B.

The cars with low speed (lower than 10 km/h) are not considered because they stop before the zebra cross if a pedestrian is still moving. A part of the measured data has been employed in ANFIS training phase. Remaining data have been employed in checking phase. The horizon of 100 epochs for the training phase has been chosen to minimize the error. The calibration process, based on ANFIS methodology with training from real data collected on field, will provide the final configuration of membership functions parameters as fixed in previous paragraph. The results, described below, are referred to the first 20 minutes of takeovers. Starting membership functions shapes were fixed as discussed in section 3, and are represented in Figure 1(a). Membership Functions obtained for Model A after the training procedure are shown in Figure 1(b).

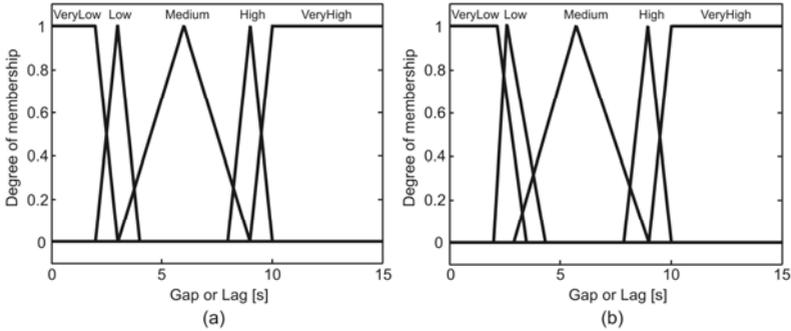


Fig. 1 Model A: Membership functions before (a) and after (b) adjustment

The positions of triangles and trapezoids vertexes have been slightly changed. The greatest shifting of vertexes is equal to 0.44 seconds for “Low” Membership Function. Even in the Model B we face a slight adjustment. The greatest vertexes shifting has been obtained in both cases for “Low” Membership Functions: 0.40 seconds for gap/lag and 0.52 kilometres per hour for speed.

5 Model Results

Output function values very close to 1 point to a positive crossing decision (lag/gap greater than about 4.3 seconds), while values very close to 0 indicate that for corresponding gap values (smaller than 2 seconds), the pedestrian decides not to cross. For values between 2 and 4.3 seconds the decision is no longer unique (the higher the preference the more the pedestrians accept the relative gap). In this interval pedestrians behave in different ways. We find a typically fuzzy situation in which the choice to cross or not has different degrees of preference. The output function of calibrated Model A is shown in Figure 2(b). As regards the Model B, the results are shown in Figure 3.

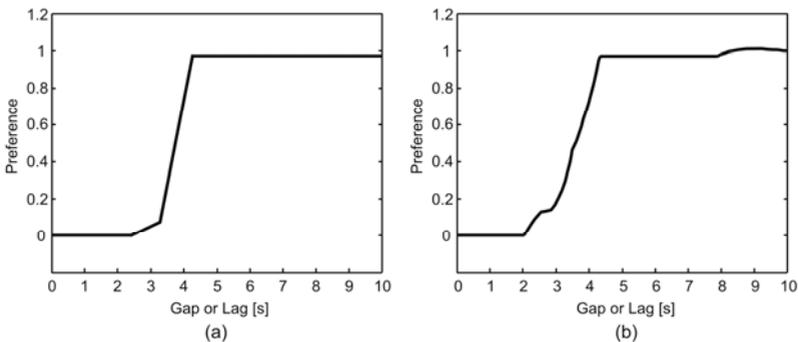


Fig. 2 Model A output function: not calibrated (a) – calibrated (b)

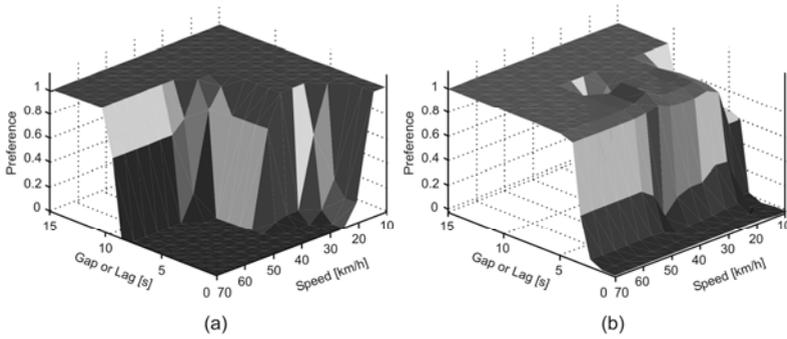


Fig. 3 Model B output function: not calibrated (a) – calibrated (b)

Unlike Model B output previously defined on the basis of expert knowledge, in this case (Model B calibrated) it can be noticed that pedestrians decision seems to be not affected by vehicles perceived speed.

6 Model Validation and Result Analysis

As well as the checking phase of the ANFIS, the models outputs were compared with sample data. While real case choices data are represented with a binary condition (1 cross – 0 not cross), Fuzzy models output is a degree of preference for crossing decision (the more the value of preference is high the more the pedestrians prefer to cross). Practical use of this decision-making model requires transformation of Model output values (preferences), in numerical conditions to be used in pedestrian behaviour simulators. For this reason, Fuzzy models output has been first normalized bringing preferences values between 0 and 1. Then to compare Fuzzy models output with measured data, (as proposed in [9, 14, 15]), it was assumed that for preferences values greater than or equal to 0.5 the pedestrian chooses to cross; on the other hand for preferences values less than 0.5 the pedestrian chooses to wait. The error assessment was conducted by splitting time and speed intervals into sub-intervals. Comparison results are shown in Table 1. For the Model A the total error is of $11/164 = 6.7\%$. For The Model B the total error is of $10/175 = 5.7\%$.

Table 1 Comparison between Model A/Model B output and real data

gap/lag [s]	Model A correct	Model A incorrect	Model A error (%)	Model B correct	Model B incorrect	Model B error (%)
< 3	76	0	0	77	0	0
3– 5	44	11	20	46	10	17.9
> 5	33	0	0	42	0	0

It can be noticed that the proposed models provide more incorrect decisions when parameter values imply subjectivity in pedestrians assessment. Anyway, total error rates obtained are widely acceptable (error < 6%), and allow to affirm that the model reproduces the phenomenon with good reliability.

The total error rate obtained with Model A and Model B have almost the same magnitude. This fact would seem to prefer the Model A, which, while providing an error rate sufficiently low, requires to collect field data for a single parameter rather than two. The results obtained in Model A and in Model B are comparable with those showed in precedent gap acceptance studies [15].

7 Conclusion and Further Research

The results of this work illustrate that fuzzy philosophy is usefully suitable to description of pedestrian decision behaviour at a crossing, as it manages to capture more accurately the subjectivity of the decision taken by the pedestrian and provides a sufficiently synthetic and, at the same time, mathematically valid description of phenomenon. The proposed model allows to reproduce human approximate reasoning, providing reliable and useful results within a soft computing environment and/or with low data requirements.

The main difference with the classic gap acceptance theory lies in the fact that the assessment performed by a pedestrian with respect to his characteristics related to ECT (walking speed and aggressiveness factor) is implicit in FIS modes. Moreover, through an ANFIS learning procedure, “ex post” evaluation procedure of pedestrian choices allows to automatically take into account evaluation and perception mistakes, inevitably made by the pedestrian.

Next steps of the research will involve the construction of Fuzzy models corresponding to other combinations of input significant parameters. A comparison between the results provided by this set of models will determine which are the parameters actually significant for the fuzzy representation of the phenomenon and will lead to outline a definitive architecture for a Fuzzy model reproduction of pedestrians behaviour at a road crossing. Then the robustness of this model will be tested on other pedestrian crossings.

References

- [1] Eurotest, Pedestrian crossings survey in Europe (January 2008), <http://www.eurotestmobility.net> (Accessed April 30, 2009)
- [2] Brilon, W., Koeniga, R., Troutbeck, R.J.: Useful estimation procedures for critical gaps. *Transportation Research Part A* 33, 161–186 (1999)
- [3] Chakroborty, P., Kikuchi, S.: Calibrating the membership functions of the fuzzy inference system: instantiated by car-following data. *Transportation Research Part C* 11, 91–119 (2003)
- [4] Chu, X., Baltes, M.R.: Pedestrian Mid-block Crossing Difficulty National Technical Information Service (NTIS), <http://www.nctr.usf.edu> (Accessed April 30, 2009)

- [5] Palamarthy, S., Mahmassani, H.S., Machemehl, R.B.: Models of Pedestrian Crossing Behavior at Signalized Intersections, Research Report 1296-1, Center for Transportation Research, University of Texas at Austin (1994)
- [6] Yang, J., Deng, W., Wang, J., et al.: Modeling pedestrians road crossing behavior in traffic system micro-simulation in China. *Transportation Research Part A* 40, 280–290 (2006)
- [7] Lobjois, R., Cavallo, V.: Age-related differences in street-crossing decisions: The effects of vehicle speed and time constraints on gap selection in an estimation task. *Accident Analysis and Prevention* 39, 934–943 (2007)
- [8] Ottomanelli, M., Iannucci, G.: Modelling pedestrian crossing performances through discrete events based simulation. In: *Proceedings of ITS-ILS07*, Cracow, Poland (October 2007)
- [9] Brewer, M.A., Fitzpatrick, K., Whitacre, J.A., et al.: Exploration of Pedestrian Gap-Acceptance Behavior at Selected Locations, *Transportation Research Record. Journal of the Transportation Research Board* 1982, 132–140 (2006)
- [10] Miller, A.J.: Nine estimators for gap-acceptance parameters. In: *Proceedings of the International Symposium on the Theory of Traffic Flow and Transportation*, Berkeley, California, June 1971. Elsevier, Amsterdam (1971)
- [11] Tian, Z., Vandeheya, M., Robinson, B.W., et al.: Implementing the maximum likelihood methodology to measure a driver's critical gap. *Transportation Research Part A* 33, 187–197 (1999)
- [12] Troutbeck, R.J.: Estimating the critical acceptance gap from traffic movements. Physical infrastructure centre, Queensland University of Technology. *Research Report* 92-5 (1992)
- [13] Das, S., Manski, C.F., Manuszak, M.D.: Walk or wait? an empirical analysis of street crossing decisions. *Journal of Applied Economy* n° 20, 529–548 (2005)
- [14] Rossi, R., Meneguzzer, C.: L'uso di osservazioni sperimentali per l'identificazione di modelli fuzzy del comportamento di gap-acceptance. In: Angeli, F. (ed.) *Metodi e tecnologie dell'ingegneria dei trasporti. Seminario 2000*, pp. 199–213 (2002)
- [15] Meneguzzer, C., Rossi, R.: Modelli fuzzy per la simulazione dei comportamenti di Gap Acceptance nelle intersezioni stradali a priorità, *Trasporti e Trazione* n. 1, 2-11 (2001)

A Cellular Genetic Algorithm for Solving the Vehicle Routing Problem with Time Windows

Iman Kamkar, Mahdiah Poostchi, and Mohammad Reza Akbarzadeh Totonchi

Abstract. Cellular Genetic Algorithms (cGAs) are a subclass of Genetic Algorithms (GAs) in which the population diversity and exploration are enhanced thanks to the existence of small overlapped neighborhoods. Such kind of structured algorithms is well suited for complex problems. In this paper, a cGA for solving the vehicle routing problem with time windows (VRPTW) is proposed. The benchmark of Solomon is selected for testing the proposed cGAs, and compares them with some other heuristics in the literature. The results demonstrate that the proposed cGA has a good potential for solving this kind of problems.

1 Introduction

The problem we address is the Vehicle Routing Problem with Time Windows (VRPTW). The VRPTW involves routing a fleet of vehicles, with limited capacities and travel times, from a central depot to a set of geographically dispersed customers with known demands within specified time windows. The time windows are two-sided, meaning that a customer must be serviced at or after its earliest time and before its latest time. If a vehicle reaches a customer before the earliest time it results in idle or waiting time. A vehicle that reaches a customer after the latest time is tardy. A service time is also associated with servicing each customer.

Iman Kamkar

Department of Computer Science, Islamic Azad University, Mashhad Branch, Iran
e-mail: kamkar.iman@gmail.com

Mahdiah Poostchi

Department of Computer Engineering, Iran University of Science and Technology,
Tehran, Iran
e-mail: ma_poostchi@comp.iust.ac.ir

Mohammad Reza Akbarzadeh Totonchi

Departments of Electrical Engineering and Computer Engineering,
Ferdowsi University of Mashhad, Iran
e-mail: Akbarzadeh@ieee.org

The route cost of a vehicle is the total of the traveling time (proportional to the distance), waiting time and service time taken to visit a set of customers. The VRPTW arises in a wide array of practical decision making problems. The current survey of vehicle routing methodologies is available in [1] and [4]. Solomon and Desrosiers [7] provide an excellent survey on vehicle routing with time windows.

In this paper, a structured type of genetic algorithms, called cellular genetic algorithm (cGA) is used for solving VRPTW. The concept of cellular genetic algorithms was proposed by Whitley [9]. In cellular genetic algorithms, the population is structured in such a way that individuals can only interact with whom located in the immediate vicinity of their locations

2 Mathematical Formulations for VRPTW

The VRPTW is represented by a fleet of identical vehicles (denoted V), and a directed graph $G = (C, A)$, which consists of a set of customers, C . The nodes 0 and $n + 1$ represent the depot, i.e., exiting depot, and returning depot respectively. The set of n vertices denoting customers is denoted N . The arc set A denotes all possible connections between the nodes (including node denoting depot). No arc terminates at node 0 and no arc originates at node $n + 1$ and all routes start at 0 and end at $n + 1$. We associate a cost c_{ij} and a time t_{ij} with each arc $(i, j) \in A$ of the routing network. The travel time t_{ij} may include service time at customer i . Each vehicle has a capacity limit q and each customer i , has a demand d_i , $i \in C$. Each customer i has a time window, $[a_i, b_i]$, where a_i and b_i are the respective opening time and closing times of i . A vehicle may arrive before the beginning of the time window meaning incur waiting time until service is possible. However, no vehicle may arrive past the closure of a given time interval, b_i . Vehicles must also leave the depot within the depot time window $[a_0, b_0]$ and must return before or at time b_{n+1} . Assuming waiting time is permitted at no cost, we may assume that $a_0 = b_0 = 0$; that is, all routes start at time 0.

The VRPTW model can be mathematically formulated as shown in fig. ?. The objective function (1) states that cost should be minimized. The constraint set (2) states that each customer must be visited exactly once by one vehicle, and constraint set (3) states that the vehicle capacity should not be exceeded. The next set of constraints (4), (5) and (6) give the flow constraints that ensure that each vehicle leaves depot 0, departs from a customer it visited and finally returns to the depot, given by node $n + 1$. The nonlinear inequality (7) (which can be easily linearized, see [1]) states that a vehicle K cannot arrive at j before $s_{ik} + t_{ij}$ if it travels from i to j . Constraint (8) ensures that time windows are observed and (9) gives the set of integrality constraints.

3 Cellular Genetic Algorithms for VRPTW

In genetic algorithm a population of individuals representing tentative solutions is maintained. Cellular GAs are a subclass of GAs in which the population is

$\min \sum_{k \in V} \sum_{i \in N} \sum_{j \in N} C_{ij} x_{ijk} \quad \text{such that;} \quad (1)$
$\sum_{k \in V} \sum_{j \in N} x_{ijk} \quad \forall i \in C \quad (2)$
$\sum_{i \in C} d_i \sum_{j \in N} x_{ijk} \leq q \quad \forall k \in V \quad (3)$
$\sum_{j \in N} x_{0,jk} = 1 \quad \forall k \in V \quad (4)$
$\sum_{i \in N} x_{ihk} - \sum_{j \in N} x_{hjk} = 0 \quad \forall h \in C, \quad (5)$
$\forall k \in V$
$\sum_{i \in N} x_{i,n+1,k} = 1 \quad \forall k \in V \quad (6)$
$s_{ik} + t_{ij} - K(1 - x_{ijk}) \leq s_{jk}$
$\forall i \in N, \forall j \in N, \forall k \in V$
$ot_i \leq s_{ik} \leq ct_i \quad \forall i \in N, \forall k \in V \quad (7)$
$x_{ijk} \in \{0,1\} \quad \forall i \in N, j \in N, \forall k \in V \quad (8)$
$V = \{1,2,\dots,k\} \quad \text{vehicles} \quad (9)$
$C = \{1,2,\dots,n\} \quad \text{customer size}$
$0, n+1 \quad \text{depot}$
$N = \{0,1,\dots,n, n+1\} \quad \text{node size}$
$d_i \quad \text{demand of client } i$
$a_i \quad \text{open time of client } i$
$b_i \quad \text{close time of client } i$
$q_k \quad \text{capacity of vehicle } k$
$t_{ij} \quad \text{time from client } i \text{ to } j$
$s_{ik} \quad \text{customer } i \text{ take } k \text{ service time}$

Fig. 1 Mathematical fomulation of VRPTW

structured in a specified topology, so that individuals may only interact with their neighbors. These overlapped small neighborhoods help in exploring the search space because the induced slow diffusion of solutions through the population provides a kind of exploration, while exploitation takes place inside each neighborhood by

genetic operations. In Figure 2 we can see a pseudo code of our basic cGA. As it can be seen, it starts by generating and evaluating an initial population. After that, the genetic operators (selection, recombination, mutation, and replacement) are iteratively applied to each individual until the termination condition is met. As mentioned before in cellular model these genetic operators are applied within the neighborhood of the individuals, so individuals belonging to different neighborhoods are not allowed interacting.

Representation of Individuals. For the representation of individuals, integer string of length N is used, where N is the number of customers. Each gene in the chromosome is the integer node number assigned to that customer originally. And the sequence of the genes in the chromosome is the order of visiting these customers.

Selection. In order to select parents for mating and reproduction, the tournament selection mechanism is used. In this scheme, the probability to select an individual is proportional to its fitness. An individual fitness is computed as follows:

$$fitness_i = N_{ri} - N_{rm} + \alpha d_i \quad (10)$$

Where:

N_{ri} = number of routes in solution i

N_{rm} = number of routes in the best solution of current population

d_i = total distance traveled in solution i

α = a parameter which is used to privilege solutions having smaller number of routes. $\alpha = 1/d_m$, where d_m is the maximum traveled distance over the individuals of the initial population.

Reproduction. The performance of GA is highly dependent to the reproduction which combines the useful traits form parent chromosomes and passes them on to the offspring. The reproduction of GA consists of crossover and mutation. It must notice that in the cellular GA, chromosomes only interact with their neighbors. This paper employs PMX crossover, The PMX proceed by choosing by two cut points by random:

Parent 1: *h k c e f d b l a i g j*

Parent 2: *a b c d e f g h i j k l*

The cut-off section defines a series of swapping operations to be performed on the second parent. In the above, we swap b with g , l with h , and a with i , and end up with the following offspring:

Offspring 1: *i g c d e f b l a j k h*

Performing similar swapping on the first parent gives the other offspring:

Offspring 2: *l k c e f d g h i a b j*

```

proc Steps(cga) // Parameters of the algorithm in 'cga'

  GenerateInitialPopulation(cga.pop);

  FitnessEvaluation(cga.pop);

  while ! StopCondition() do

    for individual ← 1 to cga.popSize do

      neighbors ← CalculateNeighborhood (cga,location(individual));

      parents ← Selection(neighbors);

      offspring ← Crossover(cga.Pc,parents);

      offspring ← Mutation(cga.Pm,offspring);

      FitnessEvaluation(offspring);

      Replacement(location(individual),auxiliary pop,offspring);

    end for

    cga.pop ← auxiliary pop;

  end while

```

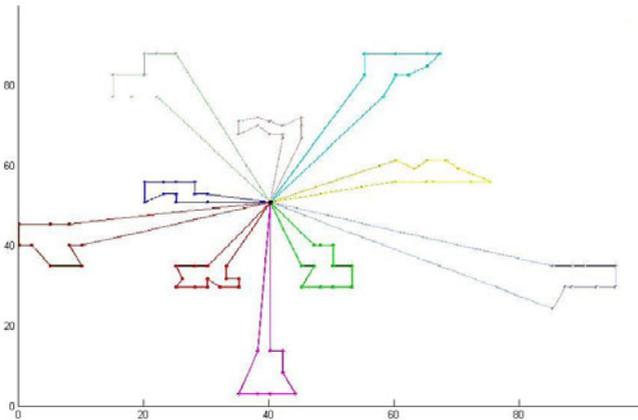
Fig. 2 Pseudo code of canonical cGA

Mutation. Here we have used “inversion” mutation, two cut points are selected in the chromosome, and the genetic material between these two cut points is reversed. Maintaining the route time window imposes major constraints since the violation of individual customer time windows can segment that route into multiple routes. Thus we employ a constrained reversal, which is limited in length to 2 customers.

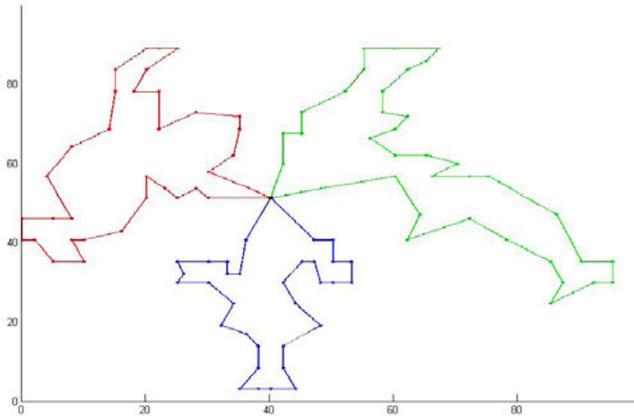
4 Experimental Results and Comparisons

In this section, we present computational results of our proposed GA, which is coded in Matlab 7.5 and executed on a personal computer with a Pentium processor running at 1GHZ. Our experimental results use the standard Solomon’s VRPTW benchmark problem instances available at [10]. Solomon’s data is clustered into six classes; C1, C2, R1, R2, RC1 and RC2. Problems in the C category means the problem is clustered, that is, customers are clustered either geographically or

according to time windows. Problems in category R mean the customer locations are uniformly distributed whereas those in category RC imply hybrid problems with mixed characteristics from both C and R. Furthermore, for C1, R1 and RC1 problem sets, the time window is narrow for the depot, hence only a few customers can be served by one vehicle. Conversely, the remaining problem sets have wider time windows hence many customers can be served by main vehicles. In Figs. 3, we illustrate some of the network topologies obtained after running the CGA for 200 generations. In figure 3-a, customers are clustered together and have a small time window. In figure 3-b, customers are also clustered but have a wider time window



(a)



(b)

Fig. 3 a) Network topology for 100 geographically clustered customers with a narrow time window, Test problem *c101* , b) Network topology for 100 uniformly distributed customers with a narrow time window. Test problem *r108* : [10].

window so the network topology shows that one vehicle can serve more customers. The comparison between the average of the best solutions obtained by CGA and other algorithms were done in table 1. Table 1 includes ant colony algorithm of Gambardella and Tillard, (1999,MACS-VRPTW) [3], methods of Rochat and Tillard, (1995,RT) [7], Tillard et al., (1997,TB) [8], Chiang and Russel (CR,1993) [2], genetic algorithm of Potvin and Bengio (PB,1996) [5], genetic algorithm of Berger et al.[]. In this table, best results are shown in boldface. It is illustrated that in all types of problems cGA outperforms other algorithms.

Table 1 Average of the best solutions computed by different VRPTW algorithms. Best results are in boldface. MACS-VRPTW= Gambardella and Tillard (1999), RT=Rochat and Taillard (1995), TB= Taillard et al. (1997), CR=Chiang and Russel (1993), PB=Potvin and Bengio (1996), GA= Berger et al.(2001).

	R1		C1		RC1		R2		C2		RC2	
	VEI	DIST	VEI	DIST	VEI	DIST	VEI	DIST	VEI	DIST	VEI	DIST
MACS-VRPTW	12.00	1217.73	10.00	828.38	11.63	1383.02	2.73	967.35	3.00	589.86	3.25	1130.19
RT	12.25	1208.50	10.00	828.38	11.88	1377.26	2.91	960.37	3.00	589.86	3.38	1119.59
TB	12.17	1209.35	10.00	828.38	11.50	1389.22	2.82	981.02	3.00	589.86	3.38	1117.34
CR	12.42	1288.76	10.00	884.76	12.38	1455.62	2.91	1134.87	3.00	657.98	3.38	1361.26
PB	12.58	1294.27	10.00	839.92	12.13	1446.24	3.00	1118.25	3.00	591.07	3.38	1229.76
GA	12.17	1251.40	10.00	828.50	11.88	1414.86	2.73	1056.59	3.00	590.06	3.25	1258.15
CGA	12.00	1215.25	10.00	828.38	11.50	1376.54	2.73	960.15	3.00	589.86	3.25	1118.39

5 Conclusion

This paper presents a cellular genetic algorithm for solving the vehicle routing problem. The used cellular GA technique maintains population diversity for more time with respect to panmictic (single population) GAs due to the use of small overlapped neighborhoods. This feature frequently prevents the algorithm from getting stuck in local optima. The proposed algorithm is compared with different algorithms proposed for solving vehicle routing problem with time windows. Results show that cGA works better than these algorithms.

References

1. Berger, J., Barkaoui, M., Bräysy, O.: A parallel Hybrid Genetic Algorithm for the Vehicle Routing Problem with Time Windows, working paper, Defence Research Establishment Valcartier, Canada (2001)

2. Bodin, L., Golden, B., Assad, A., Ball, M.: The State of the Art in the Routing and Scheduling of Vehicles and Crews. *Computers and Operations Research* 10(2), 63–211 (1983)
3. Chiang, W.C., Russel, R.: Hybrid Heuristics for the Vehicle Routing Problem with Time Windows, Working Paper, Department of Quantitative Methods, University of Tulsa, OK, USA (1993)
4. Gambardella, L.M., Tillard, E., Agazzi, G.: MACS-VRPTW: a multiple ant colony system for vehicle routing problems with time windows. *Mcgraw-Hill'S Advanced Topics In Computer Science Series*, pp. 63–67 (1999)
5. Golden, B., Assad, A. (eds.): *Vehicle Routing: Methods and Studies*. North Holland, Amsterdam (1988)
6. Potvin, J.-Y., Bengio, S.: The Vehicle Routing Problem with Time Windows - Part II: Genetic Search. *INFORMS Journal of Computing* 8, 165–172 (1996)
7. Rochat, Y., Taillard, É.D.: Probabilistic Diversification and Intensification in Local Search for Vehicle Routing. *Journal of Heuristics* 1, 147–167 (1995)
8. Solomon, M.M., Desrosiers, J.: Time Window Constrained Routing and Scheduling Problems: A Survey. *Transportation Science* 22(1), 1–11 (1986)
9. Taillard, É.D., Badeau, P., Gendreau, M., Guertin, F., Potvin, J.-Y.: A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science* 31, 170–186 (1997)
10. Whitley, D.: Cellular genetic algorithms. In: Forrest, S. (ed.) *Proceedings of the 5th ICGA*, p. 658. Morgan-Kaufmann, CA (1993)
11. <http://w.cba.neu.edu/~msolomon/problems.htm>

Fuzzy C-Means Based Stop Line and Crosswalk Placement at Right Turn Vehicle Exclusive Lane

Fengxiang Qiao, Qing Zhu, Po-Hsien Kuo, and Ying Li, and Lei Yu

Abstract. An intersection is one of the most complex traffic situations that motorists, pedestrians and bicyclists all encounter. With different crossing and entering movements by motor vehicles, pedestrians and bicycles are often facing safety problems especially when crossing the right turn vehicle exclusive lanes. Currently the pedestrian crosswalks on pavements are not properly placed and there is no detailed guideline or procedures to locate such lane markings. In this paper, a fuzzy c-means based procedure is proposed to determine the placement of pedestrian crosswalk at right turn vehicle exclusive lane, which is based on the identification of the clusters of vehicle stopping positions. A case study in La Port, Texas, USA was conducted to illustrate how the procedure can be used in the real practice.

1 Background of Research

A roadway intersection is naturally regarded as a place of traffic conflict in the roadway system. Each year pedestrian fatalities comprise approximately 11 percent of all traffic fatalities with about 4,600 pedestrian deaths, and another 70,000 pedestrians injured in roadway crashes annually in the United States (US) (FHWA 2009). The endangerment of pedestrians and bicyclists by right-turn vehicles is among one of the big concerns of intersection safety throughout the world. Furthermore, many countries (with right-hand traffic), have a right turn on red (ROR) policy, which permits vehicles to turn right at a red traffic light (almost always after a complete stop) when the way is clear. This improves the traffic efficiency of the intersection but decreases the safety level of pedestrians and bicyclists to some extent.

Fengxiang Qiao

Department of Transportation Studies, Texas Southern University,
3100 Cleburne Avenue, Houston, Texas, U.S.A.

Tel.: (713) 313-1915; Fax: (713) 313-1856

e-mail: qiao_fg@tsu.edu

<http://transportation.tsu.edu>

Pedestrian and bicycle safety improvements depend on an integrated approach that involves the so-called 4 E's: Engineering, Enforcement, Education, and Emergency Services. Among all these, the engineering treatment is one of the most basic and efficient ones. In the past decades, extensive research was conducted to protect the vulnerable pedestrians and bicycles at intersections via right-turn vehicles (e.g. Niewoehner and Berg 2005, Kikuchi and Kronprasert 2008).

The US national Manual on Uniform Traffic Control Devices (MUTCD 2003, Chapter 3B.17) establishes general guidelines on right-turn maneuvers and pedestrian/bicycle crosswalk designs. In Chapter 7C.03 of MUTCD, the crosswalk markings are defined as to provide guidance for pedestrians who are crossing roadways and to alert road users of a pedestrian crossing point across roadways not controlled by highway traffic signals or STOP signs. Crosswalks are marked at intersections where there is substantial conflict between vehicular and pedestrian movements. Figure 1 is an illustration of pedestrian crosswalks around a right turn vehicle exclusive lane.

Since the crosswalk placements are so important to traffic safety, the crosswalk markings and associated right turn vehicle stop lines should not be used indiscriminately. MUTCD suggest performing engineering study before they are installed at locations away from traffic control signals or STOP signs.

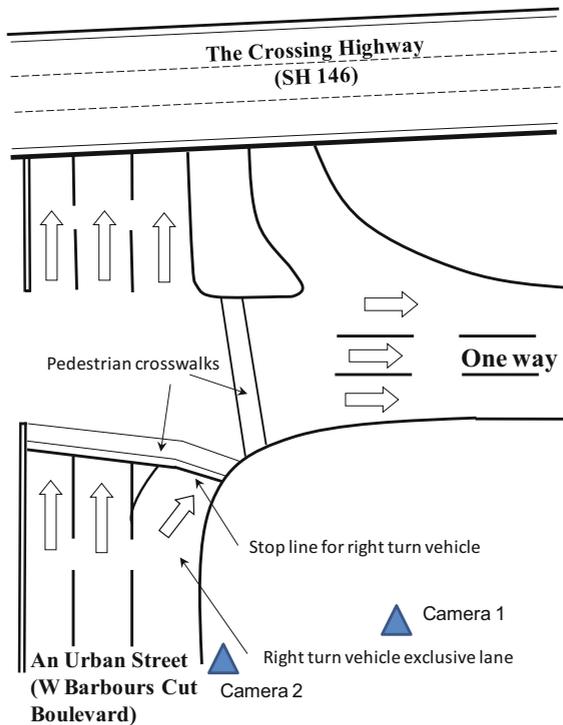


Fig. 1 Crosswalk markings around a right turn vehicle exclusive lane

In MUTCD, the width and line style of pedestrian crosswalk are clearly stated. However, due to the complexity of right turn vehicle operations, the placement of stop line for right turn vehicles in Figure 1 is not well defined. The most essential tool for use in determining crosswalk is engineering judgment (Edwards and Kelcey 2002), which is based on engineers' experiences rather than quantitative guidance. This instigates a phenomenon that most right turn vehicles do not stop behind the marked stop line, which is a safety threat to pedestrians and bicyclists crossing such right turn lanes.

2 Research Objective

The objective of this research is to develop an intelligent technical procedure to help engineers to decide where to place the right turn stop line and pedestrian crosswalks so as to improve the safety of the pedestrians and bicyclists.

3 Fuzzy *c*-Means Based Problem Formulation

3.1 Methodology Description

Like any design problems in traffic engineering, the geometric placement of pedestrian crosswalk should fully consider the vehicles' driving behaviors within the right turn exclusive lanes. Since normally the right turn vehicles are not stopping behind the assigned stop lines for pedestrian crosswalks, it is necessary to identify where most vehicles like to stop at the specific right turn lane. The placement of the crosswalk should then consider such stopping locations.

As the geometric characteristics and driving behaviors at different intersections could be different, our methodology should (1) be able to identify the cluster(s) of stopping position(s) of right turn vehicle based on the real traffic data from field observations, and (2) to select a most suitable cut-off line as the stop line for right turn vehicles.

3.2 Fuzzy *c*-Means Based Cluster Identification

The first problem in Sect 3.1 is a typical data clustering problem in pattern recognition, which is a field concerning machine recognition of meaningful regularities in noisy or complex environments. Since the right turn vehicle stopping positions are usually not well distributed, its "regularities" may not be precisely defined. To deal with such ambiguity, it would be much beneficial to introduce some "fuzziness" into the formulation of the problem (Zadeh 1965). One extra advantage is that all variables in a fuzzy model are continuous so that derivations can be computed to find the right direction for the search (Wang 1997). In the following part, we will use the most famous fuzzy clustering algorithm: the fuzzy *c*-means algorithm proposed by Bezdek (1981) to formulate the problem.

Assume the set of vehicle stopping position and its associated traffic characters (e.g. vehicle types) along the right turn exclusive lane are represented as: $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$, where n is total number of data pairs, \mathbf{x}_k is defined as:

$$\mathbf{x}_k = \begin{cases} 1, & \mathbf{x}_k \in A_i \\ 0, & \mathbf{x}_k \notin A_i \end{cases} \in R^p \tag{1}$$

In (1), p is the dimension of stopping position and other traffic characters. $k=1, 2, \dots, n$, $A_i \in P(X)$, $i=1, 2, \dots, c$, $P(\mathbf{X})$ is the set of all the subsets of \mathbf{X} .

For the fuzzy c -means algorithm, the objective is to find $U = [u_{ik}] \in M_{fc}$ and $V = (\mathbf{v}_1, \dots, \mathbf{v}_c)$ with $v_i \in R^p$ such that

$$J_m(U, V) = \sum_{k=1}^n \sum_{i=1}^c (u_{ik})^m \|\mathbf{x}_k - \mathbf{v}_i\|^2 \tag{2}$$

is minimized, where $m \in (1, \infty)$ is a weighting constant, M_{fc} is the fuzzy c -partition formulated by Bezdek (1981), $u_{ik} \in [0, 1]$ is the degree defining $\mathbf{x}_k \subseteq A_i$.

The fuzzy c -means algorithm needs to establish a necessary condition for this minimization problem.

Theorem: Let $X = \{x_1, \dots, x_n\}, x_i \in R^p$ be a given set of data. Fix $c \in \{2, 3, \dots, n-1\}$ and $m \in (1, \infty)$, and assume that $\|x_k - v_i\| \neq 0$ for all $1 \leq k \leq n$ and $1 \leq i \leq c$. Then $U = [u_{ik}]$ and $V = (v_1, \dots, v_c)$ is a local minimum for $J_m = (U, V)$ only if

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{\|x_k - v_i\|}{\|x_k - v_j\|} \right)^{\frac{2}{m-1}}}, \quad 1 \leq i \leq c, 1 \leq k \leq n, \tag{3}$$

and

$$\mathbf{v}_i = \frac{\sum_{k=1}^n (u_{ik})^m \mathbf{x}_k}{\sum_{k=1}^n (u_{ik})^m}, \quad 1 \leq i \leq c \tag{4}$$

The fuzzy c -means algorithm is based on the necessary condition (3) and (4), while the four executing steps are listed in the following (Bezdek 1981, Wang 1997).

- Step 1. For a given data set $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}, x_i \in R^p$, fix $c \in \{2, 3, \dots, n-1\}$, $m \in (1, \infty)$, and initialize $U^{(0)} \in M_{fc}$.
- Step 2. At iteration l , $l = 0, 1, 2, \dots$, compute the c mean vectors \mathbf{v}_i^l is:

$$\mathbf{v}_i^{(l)} = \frac{\sum_{k=1}^n (u_{ik}^{(l)})^m \mathbf{x}_k}{\sum_{k=1}^n (u_{ik}^{(l)})^m}, \quad 1 \leq i \leq c \tag{5}$$

Step 3. Update $U^{(l)} = [u_{ik}^{(l)}]$ to $U^{(l+1)} = [u_{ik}^{(l+1)}]$ using

$$u_{ik}^{(l+1)} = \frac{1}{\sum_{j=1}^c \left(\frac{\|x_k - v_i^{(l)}\|}{\|x_k - v_j^{(l)}\|} \right)^{\frac{2}{m-1}}}, \quad 1 \leq i \leq c, \quad 1 \leq k \leq n \tag{6}$$

Step 4. If $\|U^{(l+1)} - U^{(l)}\| < \varepsilon$, stop; otherwise, set $l = l + 1$ and go to step 2.

By following the above fuzzy c -means algorithm, the resulted cluster centers $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_c)$ can indicate the centers of stopping positions and their associated other traffic characters.

3.3 Selection of Stop Line for Right Turn Vehicles

The stopping positions defined in the cluster centers $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_c)$ cannot be directly used as the stop line of right turn vehicles. The first reason is because there could be more than one cluster center but we only need a single stop line. The second reason is that if any clusters center (or its projection on the axis “stopping position”) is directly used as the stop line, there could be still many vehicles stopping beyond the stop line. This affects pedestrians and bicycles’ crossings.

Assume that $\mathbf{sp} = (sp_1, \dots, sp_c)$ is the vector of stopping positions at the c^{th} cluster centers, which are actually the projection of $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_c)$ on the axis “stopping position”, where $sp_1 < sp_2 < \dots < sp_c$ with sp_c being the closet one to the edge of the crossing street. All the n_c stopping positions $sp_c^1, \dots, sp_c^{n_c}$ that belong to their cluster center sp_c can be represented by sp_c with a degree of the normalized distance $\{d_c^i\}$ from each position to the center sp_c .

If d_c^i ($i = 1, 2, \dots, c$) is defined as the distance between sp_c^i and sp_c . Let:

$$\hat{d}_c^i = \frac{d_c^i}{\max d_c^j}, \quad \forall i, j = 1, 2, \dots, c \tag{7}$$

Then, the set $\{\hat{d}_c^i\}$ can be regarded as the membership function of sp_c in the sense that the stopping position sp_c can represent all other stopping positions in the same cluster set.

By setting a certain threshold of the membership function th_c , $0 < th_c < 1$, the stopping position sp_{th_c} that is directly related to th_c is the needed stop line position for all right turn vehicles. The threshold th_c could be so selected that the membership function value is within a relative small range (e.g. 0.1). This range can be pre-defined by engineers based on their practical judgments.

3.4 Fuzzy *c*-Means Based Crosswalk Placement Procedure

The proposed fuzzy *c*-means based crosswalk placement procedure can be summarized into the following five steps.

- Step 1. **Data Collection.** In this step, the field data should be collected through proper data collection method(s) such as video recording or automatic traffic detection.
- Step 2. **Data Preparation.** The required vehicle stopping position, vehicle type and all other necessary information that will be used later on in modeling should be retrieved and processed.
- Step 3. **Cluster Identification.** The clusters of vehicle stopping positions should be identified using the fuzzy *c*-means algorithm introduced in Sect 3.2.
- Step 4. **Determination of Membership Function of the Closest Cluster.** The selected center is the one that is closest to the crossing street, and the membership function of its “stopping position” can be determined based on the normalized distance in Sect 3.3.
- Step 5. **Selection of Stop Line.** A threshold th_c for a small membership function value should be selected so that stop line position can be identified based on the description in Sect 3.3.

4 Case Study

In order to illustrate the proposed methodology, a case study was conducted using the field data in La Port, Texas, USA to demonstrate the fuzzy *c*-mean based procedure for the placement of the stop line and pedestrian crosswalk.

4.1 Site Selection

After several rounds of site visits, the selected site for case study is at the Texas State Highway SH 146 and the urban street W Barbours Cut Boulevard in City of La Porte of Texas, USA within the Houston metropolitan area (Figure 1). Immediately to the east of this site, there exists the Barbours Cut terminal of Port of Houston with versatility in vehicle types at this location. This would help exam the influence of vehicle type on right turn stopping positions and better determine the suitable stop line and crosswalk locations.

4.2 Data Collection and Preparation

Four field surveyors, divided into two groups conducted this survey. The equipments used for data collection include two video cameras, one tripod and necessary field measurement instruments.

Videos were recorded on a regular weekday (September 29, 2009) from 10am to 2pm. Two video cameras were placed at two sides of the right-turn lane to cover different angles (Figure 1). Tripod was used with camera 1 standing on the curb to record the stopping position of the right-turn vehicles; while camera 2 was placed inside a car to cover the whole right-turn scene.

The right turn vehicle stopping positions SP and vehicle types VT (passenger car, truck and bus, and extra long truck) were retrieved from the videos manually. A total of 305 SP-VT pairs were obtained for the use of fuzzy clustering. The zero point of stopping position (the origin) was set at the beginning of the right turn curve, while all SP values were measured based on the distance from this reference point in meters.

4.3 Cluster Identification and Stop Line Selection

The proposed fuzzy c-means based methodology was implemented in a MATLAB computer program. Less simple operations were conducted in a MS EXCEL spreadsheet. The cluster center, when projected to the axis of “stopping position” yielded a value of 23.16 meters, which is a distance from the reference point, while the membership function MF of this center sp_1 was calculated and illustrated in Figure 2.

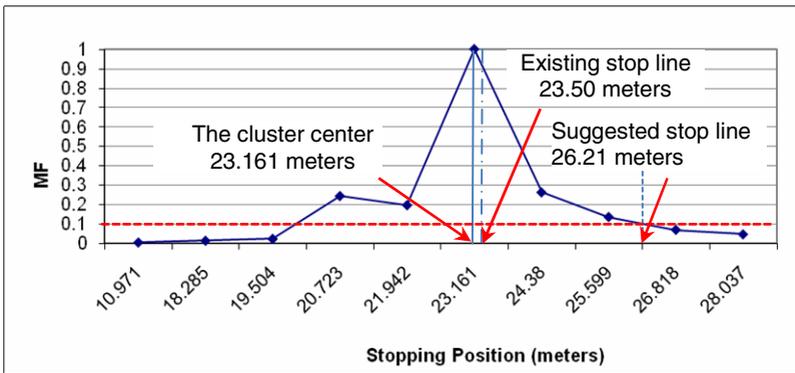


Fig. 2 Membership function MF of the vehicle stopping position for the resulted cluster center

In Figure 2, MF values were based on the occurrence rate of different vehicle stopping positions. With the hypothesized MF range selected as $th_c = 0.1$, the position of stop line for pedestrian crosswalk was identified as 26.21 meters, which is about 2.71 meters ahead of the existing right turn stop line (23.50 meters).

5 Conclusion

In this paper, the fuzzy c-means based stop line and crosswalk placement methodology is proposed and a case study in La Porte of Texas, USA was conducted to illustrate the executive procedure. A MATLAB program was compiled to automatically determine the right turn stop line position as long as the required input data pairs (actual vehicle stopping position and vehicle types) are provided. This procedure and associated program provide a feasible tool for engineers to scientifically place the right turn stop line and the related crosswalk markings. For different right turn locations with different geometrics and vehicle type compositions, the calculated pavement marking positions could be different. In this way, the majority of right turn vehicles would no longer affect the crossings of pedestrians and bicyclists even under the ROR policy.

While in the case study, the traffic data were retrieved manually, with the many advanced traffic data detection devices and techniques employed, this design procedure could be more automatic. A database could be established with similar case studies to provide guidance for placing the right turn stop lines and crosswalks even for new intersections during the planning stage. Since the proposed fuzzy c-means based approach is the first application of smart data mining methods in stopping line design, the future works include comparing its performance with that of the other data mining methods, and the before and after conflict data analysis.

References

- Bezdek, J.C.: Pattern recognition with fuzzy objective function algorithms. Plenum Press, NY (1981)
- Edwards, Kelcey: Kane County Bicycle and Pedestrian Plan, ch.4, Pedestrian Design Guide (2002), <http://www.co.kane.il.us/DOT/COM/Bicycle/FINAL/ch04.pdf> (Accessed October 16, 2009)
- Federal Highway Administration (FHWA), Intersection Safety, <http://safety.fhwa.dot.gov/intersection/> (Accessed October 13, 2009)
- Federal Highway Administration (FHWA), Manual on Uniform Traffic Control Devices (MUTCD) – 2003 Edition Revision 1 (2003), <http://mutcd.fhwa.dot.gov/HTM/2003r1/part3/part3b2.htm> (Accessed October 16, 2009)
- Kikuchi, S., Kronprasert, N.: Determining Length of Right-Turn Lane at a Signalized Intersection. Transportation Research Record: Journal of Transportation Research Board (2060) (2008), ISSN-0361-1981

- Niewoehner, W., Berg, F.A.: Endangerment of pedestrians and bicyclists at intersections by right-turn trucks. National Highway Traffic Safety Administration. In: Proceedings of 19th International Technical Conference on the Enhanced Safety of Vehicles, Washington DC (June 2005), <http://www-nrd.nhtsa.dot.gov/pdf/nrd-01/esv/esv19/05-0344-0.pdf> (Accessed October 18, 2009)
- Zadeh, L.A.: Fuzzy Sets. *Information and Control* 8, 338–353 (1965)
- Wang, L.X.: *A Course in Fuzzy Systems and Control*. Prentice Hall PTR, Upper Saddle River (1997c)

Part VIII
Surveys and Tutorials

New Trends of Soft Computing Methods for Industrial and Biological Processes

Bernardo Penna Resende de Carvalho and Leonardo Carneiro de Araújo

Abstract. This work presents real world examples of using different Soft Computing methods in both industrial and biological processes from the years 2005 to 2009. Multi-Objective Algorithm, Least Squares Support Vector Machine and Fuzzy Inference were applied in steel industry processes, while Decision Tree, Recursive Feature Elimination and Genetic Programming were evaluated in biological processes. Soft Computing methods were capable to predict quantities, recognize patterns and select relevant attributes in order to improve each process. This paper shows the growing development on Soft Computing and the integration of process knowledge points to a direction of increasing possibilities to achieve better performances in industrial and biological processes.

1 Introduction

It's widely known that industries are an essential component of sustainable development of the economy. On the other hand, industries often deal with a degradative process which can result in pollution and exhaust of natural resources. Biological processes play an important role in the society nowadays for the development of innovative treatments and medicines, but the development of drugs is still a slow process which demands thousands of biological tests on different organisms, which leads to the death of many animals in order to understand the action mechanism of proteins and other molecules.

Bernardo Penna Resende de Carvalho
Optimization Department, ABG Information Systems, Brazil
e-mail: bpenna@gmail.com

Leonardo Carneiro de Araújo
CEFALA, Federal University of Minas Gerais, Brazil
e-mail: leolca@gmail.com

At those points of opposing interests rises the necessity to optimize the processes in order to achieve the best usage of the available resources, which can be done using state-of-art technologies available. In the last two decades, advances have been made in the technologies of Soft Computing. Beyond classical methods [8, 9, 10, 11], many techniques were recently developed [7, 12, 13, 14] and are proven to give good results in the fields of pattern recognition [1, 4], synchronization [5, 6] and prediction [2, 3] of process variables.

Steel industries need not only to produce their products with the required quality, but also to ensure that their production process is performed using the best practices with ecological sustainability. This includes the higher final product quality, the higher production speed and also the lower and efficient energy and raw-materials consumption.

Soft Computing methods are used in biological industries in order to help the development of new drugs and treatments, as well as reducing the market entrance time of a new medicine or even making possible some processes previously impracticable. This can be possible because *in silico* tests are much faster to be carried through than *in vivo* ones.

This work aims at giving real world examples of using different Soft Computing methods in both industrial and biological processes developed during the years 2005 to 2009. Multi-Objective, Least Squares Support Vector Machine and Fuzzy Inference were applied in steel industry processes (Sect. 2), while Decision Tree, Recursive Feature Elimination and Genetic Programming were evaluated in biological processes (Sect. 3). It can be observed in Sect. 4 that the usage of these technologies on real industrial and biological cases made possible to predict quantities, recognize patterns and select relevant attributes in order to improve each process.

2 Applications of Soft Computing on Steel Industrial Processes

Soft Computing methods provide the capability to generate knowledge directly from raw industrial data, which improved the know-how and the productivity of industrial processes. In this section we present steel industry cases where Multi-Objective Algorithm, Least Squares Support Vector Machine and Fuzzy Inference were successful implemented in optimization systems from 2005 to 2007.

2.1 Multi-objective Algorithm

Multi-Objective Algorithm (MOBJ) [13] is a learning scheme for improving generalization of MLPs (Multilayer Perceptrons) [8], which are based on a connectionist approach to computation composed by building blocks, the neurons, interconnected as a network. Each neuron of MLP is a computing unit that weights its input data and provides an output according to its transfer function. MOBJ minimizes both sum of squared error and norm of network weight vectors to obtain the Pareto-optimal solutions, which are not unique. After the usage of a validation set, the final solution

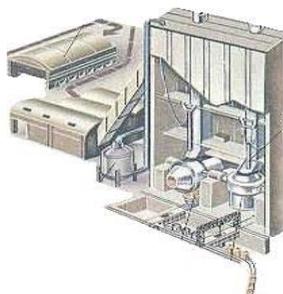


Fig. 1 Blast furnace and converter



Fig. 2 MOBJ based system

is expected to balance network variance and bias. Since the training parameters produce minor effects on the final solution, tuning the best set of training parameters is an easy task. MOBJ's result is the best MLP generated with the Pareto-optimal solutions, where the criterion is the validation accuracy.

The steel processing depends on the synchronization of temperature in both the converter and the blast furnace, presented on Fig. 1 in order to achieve a desired temperature at the continuous casting. The system presented in Fig. 2 was developed to control temperature in the steel process of the blast furnace. It helps the operator to take decisions, being able to predict and track the thermal behavior of the steel during its processing inside the converter, blast furnace and continuous casting. This system, which is based on 14 variables of the process, is able to predict the thermal losses from the converter to the blast furnace. It got an accuracy of 78%, when considering an error as a difference greater than 150 degrees between the real and the prediction temperature [6]. The consequences of the system were an increase in the productivity of the steelworker and a lower raw-material consumption of the hole process.

2.2 Least Squares Support Vector Machine

Least Squares Support Vector Machine (LS-SVM) [12] is a learning machine that corresponds to a modified version of Support Vector Machine (SVM) [14]. This new SVM formulation consists of using equality constraints at the primal cost function to be minimized, instead of inequality ones. Moreover, the 2-norm is used for the slack variables of the primal problem, instead of the 1-norm. As a result of these modifications, the problem generated by LS-SVM can be solved with a system of linear equations, which is less complex than the quadratic programming used in SVM. LS-SVM is a way to deal efficiently with outliers. It uses the principle of structural risk minimization, which results in a high capacity of generalization, even if the training set is not very representative. A LS-SVM drawback is the parameter tuning difficulty during the training phase.

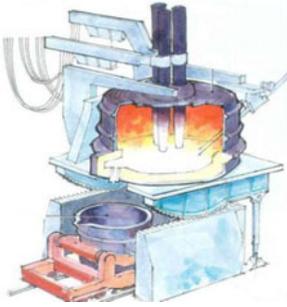


Fig. 3 Electric arc furnace

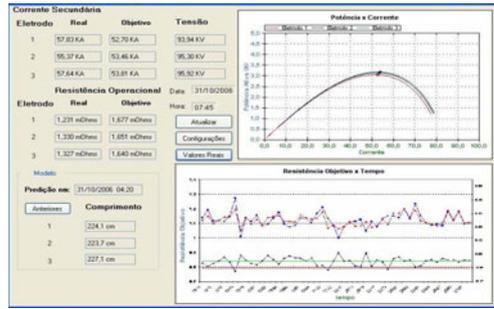


Fig. 4 LS-SVM based system

Electric arc furnace is an equipment often used for steel making with three graphite electrodes (Fig. 3). One of the most laborious and also very dangerous action in steel industry is the measurement of the electrode length. A system developed to estimate the three electrode lengths of an electric arc furnace is presented in Fig. 4. This system was capable to estimate each electrode length during the time, decreasing the frequency that the furnace has to be opened, avoiding damage on it and also increasing its productivity. It got an accuracy higher than 70% in the prediction of three electrode lengths, with a hit been considered when there is a difference lower than 15cm between the prediction and the real value [2].

2.3 Fuzzy Inference

Fuzzy Inference [9] is a method which formulates a mapping from a given input to an output using fuzzy logic instead of using sharp switching between values, dealing with the concept of approximate rather than precise. In the traditional view of sets, there are crispy limits for each set, making it possible to determine precisely either an element belongs to the set or not. In fuzzy sets, the concept of possession is rather diffuse, which implies in membership values that states the degree a certain element belongs to a set or another. Just like the fuzzy sets, in fuzzy logic a statement has a degree of truth that ranges from 0 to 1, not being constrained to only two values as in classic propositional logic. Fuzzy Inference uses the provided input information to derive inferred conclusions based on fuzzy logic and than use this result to guide the decision making using a set of if-then fuzzy rules.

A continuous casting process in steel industry is presented in Fig. 5. It's very important to control the synchronization of this process in order to allow the correct usage of the next equipments. A fuzzy system (Fig. 6) was implemented in this process phase, where fuzzy rules were used in order to estimate the production status of a casting process: on time, delayed or advanced. These rules were used to synchronize the casting phase and resulted in an advance of two runs a day and an economy of electricity consumption [5].



Fig. 5 Continuous casting process

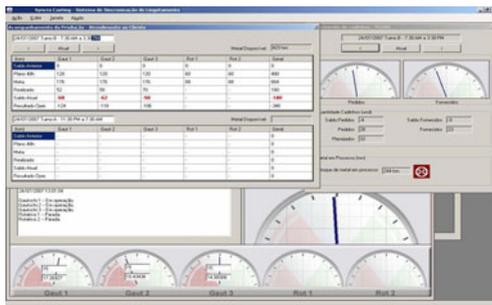


Fig. 6 Fuzzy based system

3 Applications of Soft Computing on Biological Processes

Biological and pharmaceutical industries have a straight relationship with computing and molecular biology, which aim at analyzing large amounts of biological data, predicting gene functions or demonstrating the relationships between genes and proteins in order to improve medicine development. In this section it's presented the results of Decision Tree, Recursive Feature Elimination and Genetic Programming applied into biological processes from 2006 to 2009.

3.1 Decision Tree

Decision Tree is a method [11] which uses symbolic relationships, in the form of a hierarchical tree structure, in order to classify samples based on a series of rules about their attributes. It classifies samples from the root to the leaves, which corresponds to a specific class. Each node of Decision Tree performs an attribute test for the sample, in a manner that each alternative way corresponds to a possible value for that variable. The process of walking through the tree is repeated until it reaches a leaf, where the pattern is stated for the correspondent class. It does not require any knowledge or parameter setting. The attributes of the classes can be any type of variables from binary, nominal, ordinal, and quantitative values, while the classes must be qualitative type (categorical or binary, or ordinal).

Glands play a very important role in the overall control function of organism bodies (Fig. 7). It was developed a system based on Decision Tree in order to classify samples of glands from various species based on their levels of genomic expression existent in a database of glands and non-glands. The system based on Decision Tree presented in Fig. 8 was capable to evaluate a classification between glands and non-glands with 100% accuracy using information of only 2% of the complete database of more than 4500 expression levels [12].

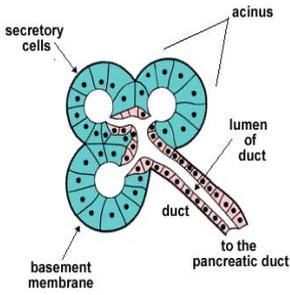


Fig. 7 A gland structure

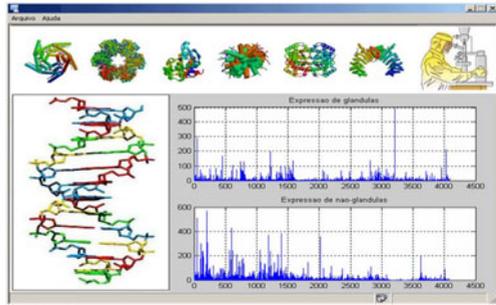


Fig. 8 Decision tree based system

3.2 Recursive Feature Elimination

Recursive Feature Elimination (RFE) [7] is a feature selection technique that uses SVM [14] to detect the most relevant features of a given database. RFE selects a subset of relevant features by using a ranking criterion that computes the variation in an objective function upon removing each feature in a SVM classifying task. In order to improve the efficiency of SVM training, this objective function is represented by a cost function for each feature and it's computed by using the training set only. When a feature is removed or its correspondent weight is reduced to zero that attribute is considered as irrelevant, because it has not any influence in building the separation hyperplane. The great advantage of RFE is the minimization of the original search space of the problem into a significantly simpler space, what makes easier the classification task.

Synapses are spaces filled with a fluid, between the neurons, which separate the emitting and receiving cells. The nervous signal is transmitted from the pre-synaptic to the post-synaptic neuron, through the synapse (Fig. 9). Some types of proteins can be found in the post-synaptic neurons, being important to the transmission of information. More interesting than only determining if a given protein possess post-synaptic activity or not, it is important to discover which are the most relevant

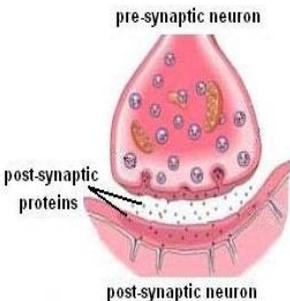


Fig. 9 A synapse structure

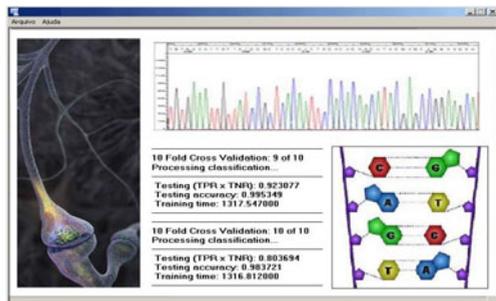


Fig. 10 RFE based system

features related to this task, which is done by the developed system (Fig. 10). It uses RFE in order to select only the relevant features from the complete protein database. Once the reduced subset was found, it was used to predict its classes. The results show that the system led to a reduced representation to the database, using only 6% of the original information, and yielded a classification accuracy of 96% [3].

3.3 Genetic Programming

Genetic Programming (GP) [10] is an evolutionary algorithm inspired by biological evolution. It aims at finding computer programs, represented by trees, that perform a given task. The evolution process is performed according to a fitness function, with new individuals being created in each generation. Every tree node (individual) has an operator function and every terminal node has an operand, making mathematical expressions easy to evolve and evaluate. There are two operators that simulate the evolutionary process: mutation and crossover. Mutation affects only one individual in the population, by replacing a whole node in the selected individual or just one node's information. Crossover is applied by simply switching one of individual nodes with another node from another individual in the population. With a tree-based representation, replacing a node means replacing the whole branch.

Some toxins are able to increase the permeation of ions through the cellular membranes, by binding to ion channels (Fig. 11). Ion channels are proteins normally involved with the functioning of the nervous system of many organisms. Fig. 12 presents a system that can store raw data from protein purification and characterization experiments and uses GP to discover novel patterns within the primary and secondary structures of a set of toxins. The system discovered patterns made possible to differentiate these toxins by their function: binding to specific channels for sodium, calcium or potassium ions. The experiments were performed using 802 toxin primary sequences labeled as channel functions. The system got an average classification accuracy of 80% for scorpion and spider toxins, with correctness of 97%, 67% and 55% respectively to sodium, potassium and calcium channels [4].

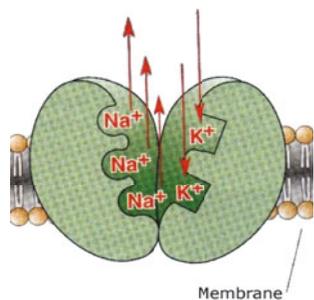


Fig. 11 An ion channel structure



Fig. 12 GP based system

4 Conclusion

Several applications of Soft Computing in both industrial and biological processes were presented in this work. All systems presented were developed, implemented and tested by the authors and colleagues. The study on Soft Computing methods and optimization techniques is a wide research area still under development. The main point of this paper is to show the growing possibility and stimulate the usage of Soft Computing to solve real world problems.

Every process has its own peculiarities and there is no rule of thumb on how to optimize it. The development of a new solution requires great knowledge of the process, what is better achieved with a collaborative work of the process specialists and the specialists in Soft Computing. This work shows successful cases where this kind of collaboration occurred.

References

1. Carvalho, B.P.R.: Classification of Gland Cells Using Decision Tree into a KOG Database. ABG Systems Technical Report. Belo Horizonte, Brazil (2006)
2. Carvalho, B.P.R., Carvalho, D.H.D., Yovanovic, A.P.: Intelligent System for Prediction of Electrode Length in an Electric Arc Furnace. In: XVI Automatic Brazilian Congress, CBA, Salvador, Brazil (2006)
3. Carvalho, B.P.R., Medeiros, T.H., Ribeiro, R.S.: Prediction of Post-Synaptic Activity in Proteins Using Recursive Feature Elimination. In: XV European Symposium on Artificial Neural Networks, ESANN, Bruges, Belgium (2007)
4. Carvalho, B.P.R., Mendes, T.M., Ribeiro, R.S., Fortuna, R., Veneroso, J.M., Mudado, M.A.: A System for Recognition of Biological Patterns in Toxins Using Computational Intelligence. In: IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, CIBCB 2009, Nashville, United States (2009)
5. Carvalho, D.H.D., Jesus, F.R., Martins, G.C.A.G.: A Fuzzy System for Casting Operation Synchronization. ABG Systems Technical Report. Belo Horizonte, Brazil (2007)
6. Carvalho, D.H.D., Maia, B.T., Moreira, A.P., Fonseca, A., Fortuna, R., Ramos, K.C.F.: Blast Furnace System for Synchronization of Operations using Computational Intelligence. In: XXXV Seminar of Melting, Refining and Solidification of Metals, ABM 2005, Brazil (2005)
7. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.N.: Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning* 46(3), 389–422 (2002)
8. Haykin, S.: *Neural Networks, A Comprehensive Foundation*. Macmillan, New York (1994)
9. Jang, J.S.R., Sun, C.T., Mizutani, E.: *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, United States (1997)
10. Koza, J.R.: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. The MIT Press, Cambridge (1992)
11. Quinlan, J.R.: Decision Trees and Multi-Valued Attributes. *Machine Intelligence* 1(1), 81–106 (1985)

12. Suykens, J.A.K., Vandewalle, J.: Least Squares Support Vector Machine Classifiers. *Neural Processing Letters* 9(3), 293–300 (1999)
13. Teixeira, R.A., Braga, A.P., Takahashi, R.H.C., Saldanha, R.R.: Recent advances in the MOBJ algorithm for training artificial neural networks. *International Journal of Neural Systems* 11(3), 265–269 (2001)
14. Vapnik, V.N., South, J., Blass, B.: *The Nature of Statistical Learning Theory*. Springer, London (1995) (2001)

Evolutionary Multi-objective Optimisation of Business Processes

Ashutosh Tiwari, Kostas Vergidis, and Chris Turner

Abstract. This paper discusses the problem of business process optimisation within a multi-objective evolutionary framework. Business process optimisation is considered as the problem of constructing feasible business process designs with optimum attribute values such as duration and cost. The proposed approach involves the application of a series of Evolutionary Multi-objective Optimisation Algorithms (EMOAs) in an attempt to generate a series of diverse optimised business process designs for the same process requirements. The proposed optimisation framework introduces a quantitative representation of business processes involving two matrices one for capturing the process design and one for calculating and evaluating the process attributes. It also introduces an algorithm that checks the feasibility of each candidate solution (i.e. process design). The experimental results demonstrate that the proposed optimisation framework is capable of producing a satisfactory number of optimised design alternatives considering the problem complexity and high rate of infeasibility.

1 Introduction and Related Work

This paper presents a business process optimisation framework (bpo^F) that is based on Evolutionary Multi objective Optimisation Algorithms (EMOAs). Business process optimisation is one of the key research areas that provide a formal perspective towards the concept of business processes. There are only a few approaches towards multi-objective optimisation of business processes which assume a fixed design for the process and optimise only the participating tasks. The utilisation of evolutionary techniques in business processes provides the advantage of working with a population of designs thus providing the capability of generating a series of diverse business process designs based on specific process

Ashutosh Tiwari · Kostas Vergidis · Chris Turner
School of Applied Sciences, Cranfield University,
Cranfield, Bedfordshire, UK
e-mail: a.tiwari@cranfield.ac.uk, c.j.turner@cranfield.ac.uk

requirements. Wang *et al.* [6] note that process optimization is a difficult task due to the non-linear, non-convex and often discontinuous nature of the mathematical models involved. Regarding business processes, the evolutionary approaches reported are limited. Hofacker and Vetschera [1] have attempted to transform and optimize a business process model using GAs but they report non satisfactory results. Tiwari *et al.* [2] and Vergidis *et al.* [4] extended this mathematical model and applied multi-objective optimization algorithms, such as the Non-Dominated Sorting Genetic Algorithm 2 (NSGA2) and the Strength Pareto Evolutionary Algorithm 2 (SPEA2) and report satisfactory results that provide encouraging opportunities for further investigation. Also, work by Vergidis *et al.* [5] introduced the concept of composite business processes. This evolutionary multi-objective optimisation approach focuses on the tasks that compose a business process rather than the business process design itself. This paper presents the extension of the optimisation framework presented by the authors in [3] that generates optimised business processes with diverse designs that are constructed based on predefined process requirements.

2 Quantitative Representation of Business Process Designs

A *business process* is perceived as a collective set of tasks that when properly connected perform a business operation. The aim of a business process is to perform a *business operation*, i.e. any service-based operation that is producing value to the organisation. The elements that are involved in the business process and consequently represented in the business process design are:

1. The participating tasks,
2. The resources of a task/business process,
3. The attributes of a task/business process and
4. The connectivity patterns.

The main elements involved are the *tasks* and *resources* of the business process. The *attributes* of the tasks and the process are also taken into consideration in order to provide the capability of evaluating a business process design. Finally, the *patterns* that interconnect the tasks are also included, as they are identified as one of the key characteristics that distinguish business processes. As task attributes we consider measurable (quantitative) characteristics of the tasks. Examples involve task cost and task duration. The task attributes can be mapped to the corresponding process attributes (e.g. process cost) using a suitable aggregate function. We consider the input and output products of a task as task resources. The nature and the type of the resources are not taken into account by the proposed framework. The task resources connect the various tasks based on common inputs and outputs. Also, the resources can shape the requirements for a process design in the form of required *process input* and expected *process output*. The set of n_d tasks that belong to a particular process design is $N_d = \{t_1, t_2, t_3, \dots, t_{n_d}\}$. The set of r_d resources in the design $R_d = \{r_1, r_2, r_3, \dots, r_{r_d}\}$ accommodates the subsets R_{in} and R_{out} that store the process input resources and process output resources respectively. The

business process design utilises all the resources in R_{in} and produces all the resources in R_{out} . Also, each task i in the design has t_{in} input resources stored in $I_i \subseteq R_d$ and t_{out} output resources stored in $O_i \subseteq R_d$. Finally, each task i has p attribute values stored in the TA_i set and the corresponding p process attributes are stored in the PA set. The participating tasks in the process design are captured using two matrices, one for capturing the task sequencing (process design) and another for capturing the task attributes. The first matrix to capture the process design itself, is a two-dimensional matrix called *Task-Resources Matrix* (TRM) and maps the input and output resources for each task. TR dimensions are $n_d \times r$ as the rows in TRM equal the number of the n_d tasks in the process design and the columns equal the number of r available resources. TRM stores the input and output resources of all the tasks in the process design, following the set of rules presented in [3]. The addition in this paper is that TRM can capture the task sequencing of a business process design and also provide a basis for reproducing one based on the business process requirements. Also, the mapping of a business process design in TRM provides the capability of setting rules in order to capture and map the various patterns that can occur in the design. The rules are based on the tasks and the way the resources flow. The resources connect the tasks and shape the various patterns that occur.

Table 1 Rules to Capture Business Process Patterns

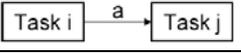
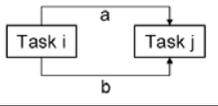
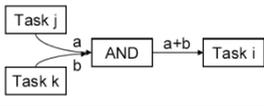
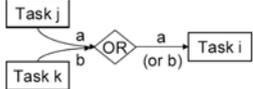
TRM MAPPING	VISUAL REPRESENTATION
SEQUENCE occurs if: $TRM_{ia} = 2$ and $TRM_{ja} = 1$	
LOOP occurs if: $TRM_{ia} = 2$ and $TRM_{jb} = 1$ AND $TRM_{jb} = 2$ and $TRM_{ia} = 1$	
AND occurs if: $TRM_{ja} = 2$ and $TRM_{ia} = 1$ AND $TRM_{kb} = 2$ and $TRM_{ib} = 1$	
OR occurs if: $TRM_{ja} = 2$ and $TRM_{ka} = 2$ AND $TRM_{ia} = 1$	

Table 1 demonstrates the rules for each pattern in TRM mapping and the corresponding visual representation. For two tasks i, j to be placed in sequence, at least an output resource of task i needs to be input resource of task j . A loop is created in the process design when for two tasks i, j one resources flows from i to j and one from j to i . Parallel execution (AND or AND-join) occurs when a task i requires different resources from (two or more) different tasks (j, k). The second matrix is similar to TRM and aims at capturing the attribute values of the tasks that

participate in the business process design. This matrix is called *Task-Attributes Matrix* (TAM) and it is used to evaluate a business process design based on the task attributes. TA dimensions are $n_d \times p$ as the rows in TAM equal the number of the n tasks in the process design and the columns equal the number p attributes per task. For an example of TRM and TAM see Vergidis and Tiwari [3].

3 Business Process Optimisation Problem

This section presents the formulation of the business process optimisation problem based on the proposed business process representation. The problem formulation assumes that there are multiple process attributes to be used as optimisation objectives. It is, therefore, a *multi-objective optimisation* problem. The multi-objective problem formulation for business process optimisation is as follows:

For a business process design with a set of n_d tasks and p process attributes:

Minimise / maximise $(PA_1, PA_2, \dots, PA_p)^T$

Subject to:

1. DoI = 0
2. $n \geq n_d > 0$
3. $r \geq r_{in}, r_{out}, t_{in}, t_{out} > 0$
4. $p \geq 2$

It should be noted that the proposed optimisation framework is oriented towards bi-objective optimisation. It is also focused on min-max problems, i.e. the minimisation of the first objective and the maximisation of the second. We assume that the process attributes are used as the optimisation objectives. A process attribute (PA_j) can be calculated as an aggregate of the corresponding task attributes stored in TAM for all the n_d tasks in the process design according to equation 1:

$$PA_j = \sum_{i=1}^{n_d} TAM_{ij} \quad (1)$$

The problem formulation involves 4 constraints. Constraint (1) ensures that only *feasible* business process designs are evaluated. The problem formulation described gives rise to some challenging issues in terms of generating optimised business process designs. The main challenges for business process optimisation are:

1. *Nature of the problem.* Based on the problem formulation, business process optimisation is a *discrete* problem as the main variable is a set of tasks (N_d) that form the business process design.
2. *Multi-objective formulation of the problem.* In addition to the discrete nature of the problem, we have assumed a multi-objective nature for the business process optimisation problem. Assuming that the participating objectives are conflicting and that each solution represents a different trade-off between the objectives, discovering the Pareto-optimal front across all of the objectives is another major challenge for the proposed optimisation framework.
3. *Solution representation.* The problem formulation requires different aspects of the business process design for different stages.

4. *Degree of Infeasibility (DoI)*. The first constraint of the problem requires the execution of the Process Composition Algorithm (PCA) algorithm. The PCA is a central part of the representation which elaborates (constructs) a business process design and also checks whether it is feasible based on the design requirements of the business process. As a result, PCA can generate alternative designs based on the same process requirements. The execution of the PCA is necessary for the measurement of the Degree of Infeasibility (DoI) for a solution. However, PCA also *updates* the solution (either removing or replacing tasks in the N_d set) in order to ensure its feasibility. This is a major challenge for the optimisation framework, to handle a solution that is modified by an algorithm during the optimisation process.
5. *Solution size*. Business process optimisation requires solutions of variable size. Having a fixed solution size and thus fixed number of tasks in the business process design would be a major barrier towards lean design composition. The framework must be capable of handling solutions of variable size for the same design requirements.

The next section introduces the proposed optimisation framework that addresses the above-mentioned challenges in order to generate optimal results in the form of alternative business process designs.

4 Proposed Optimisation Framework

The proposed business process optimisation framework (bpo^F) applies a number of existing EMOAs to a business process design captured using the proposed representation. The aim of the framework is to fully utilise the proposed representation technique and the capabilities of the EMOAs in order to generate a series of alternative optimised designs.

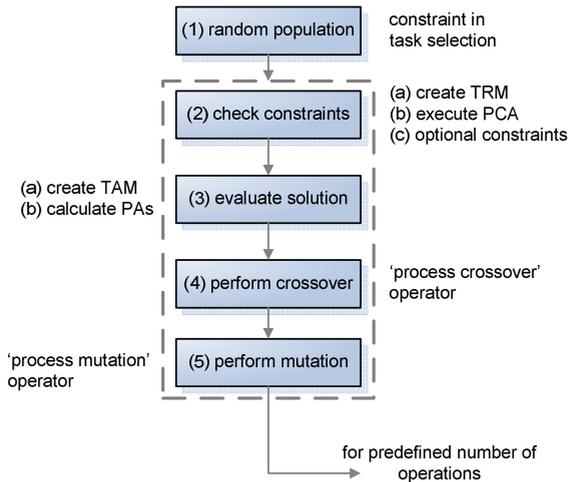


Fig. 1 bpo^F – Business Process Optimisation Framework

For each design the framework produces:

1. The *tasks in the design*, stored in the N_d set.
2. The *process graph*, which is the diagrammatic representation of the design.
3. The *Degree of Infeasibility (DoI)*, which for the optimised process designs should be equal to zero.
4. The *process attribute values*, which are calculated based on the input functions. These are the objective values which quantitatively show how well the design performs based on the criteria it has been assessed with.

The main steps and the structure of the proposed business process optimisation framework are shown in Fig. 1.

The proposed optimisation framework consists of five steps:

1. *Generate random population*: The first step of the optimisation process is the generation of random population. This step occurs only once in the optimisation process as then the population is evolved for a defined number of generations.

2. *Check constraints*: For each solution of the population, the problem constraints are checked. Note that bpo^F checks the constraints prior to solution evaluation due to a specific reason: *the constraints modify the solution*.

3. *Evaluate solution*: The solution evaluation involves two stages based on the proposed representation: (i) TAM is created and (ii) the various process attributes are calculated based on their functions. TAM is created based on updated version of the solution involving the tasks in the design and their attribute values. Based on this matrix the solution is evaluated in terms of the process attribute values.

4. *Perform crossover*: Crossover occurs directly in the N_d set of each solution. Initially, the solutions are selected for crossover based on a given crossover probability. The solutions that are chosen for crossover are split into pairs. For each pair a unique crossover-point is defined based on a random number (between 1 and n_d-1). Note that step 2 checks whether the solution is feasible.

5. *Perform mutation*: This operator randomly alters information in a chosen solution. The operator is applied on the N_d set of tasks of a particular solution. When mutation occurs a task is replaced with an *arbitrary* task from the task library.

5 Experimental Results

In order to generate satisfactory results, the EMOAs need to achieve two goals: (i) convergence to the Pareto-optimal front (in order to obtain optimal business process designs) and (ii) maintenance of the population diversity across the front (in order to obtain a variety of different sizes of business process designs). Based on these performance measures, the *features* of the problem that require further investigation are:

- A. The number of feasible solutions in a given business process
- B. The different acceptable process sizes of a feasible business process design
- C. The ranges of the task attribute values.

The work presented in this paper is focused on feature B. The experiments presented seek to investigate the *maximum size of business process designs for limited library sizes*. The outcome of this experiment is expected to suggest the maximum size of business process designs that the framework provides satisfactory results. Table 2 shows two selected scenarios from the experiments, each with different process size. The parameters that vary across the scenarios are highlighted in bold. Table 3 shows the parameters for each of the EMOAs employed by the proposed optimisation framework. Although 25,000 evaluations might seem excessively high; for most algorithms it helped produce better quality results in comparison with lower numbers and in a timely fashion. For each experiment, each algorithm is executed for 30 independent runs. The results from one of the typical results are shown in the diagrams (Fig. 2 and Fig. 3.).

Table 2 Parameter Definition for Scenarios A and B

Parameter	scenario A	scenario B
n	60	90
n_d	20	30
n_{min}	16	26
r	20	20
t_{in} / t_{out}	3	3
r_{in} / r_{out}	5 / 10	10 / 10
p	2	2
α	100 -115	100 -115
β	200 -230	200 -230

Table 3 Parameter Specification for the EMOAs Employed in bpo^F

Parameter	NSGA2	PAES
Population	500	-
Archive	-	1000
Bisections	-	5
Generations	25,000	25,000
Crossover	0.8	0.8
Mutation	0.2	0.2

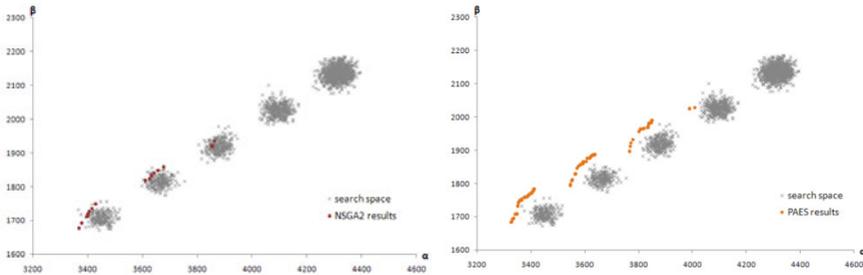


Fig. 2 Optimisation results for scenario A (NSGA2 result on the left and PAES on the right)

For scenario A NSGA2 performs poorly by identifying only 3 non-dominated solutions and taking the most time, out of both algorithms, to generate the results with 6.9 minutes per non-dominated solution. In contrast PAES generates by far the most non-dominated solutions (50) in a time of only 0.21 minutes per solution. PAES is also able to identify a non-dominated solution for $n_d = 19$ tasks (see Fig.2 for scenario A results). For scenario B the proposed optimisation framework is tested for 30-task business process designs with a library of 90 tasks. NSGA2

discovers zero non-dominated solutions while PAES demonstrates a - comparatively- very good performance. NSGA2 performs poorly identifying few feasible solutions none of which is non-dominated. PAES identifies 30 non-dominated solutions and, unlike NSGA2, can identify non-dominated solutions in the first three islands (see Fig.3. for scenario B results). It is clear PAES performs consistently well in the experimental scenarios.

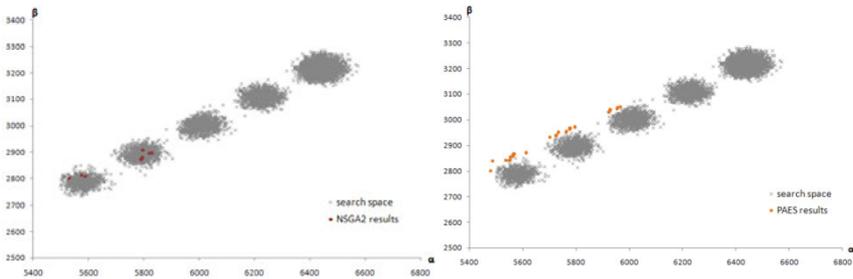


Fig. 3 Optimisation results for scenario B (NSGA2 result on the left and PAES on the right)

It discovers the most non-dominated solutions and it is also the faster of the two algorithms. PAES is strongest in cases when local search seems superior to or competitive with population-based methods. Scenario B involved large solution sizes which NSGA2 found hard to cope with. PAES using the simple (1+1) evolution strategy managed to be more effective in discovering optimised business process designs. It was also noted that with NSGA2 the performance deteriorates as the maximum process size increases.

6 Conclusions

This paper presented an extended framework for business process optimisation. The framework involved a quantitative representation for business processes, an algorithm that composes feasible process designs and a series of optimisation algorithms that generates diverse optimised designs. The results demonstrated that the framework is capable of generating diverse designs and selecting those with minimum objective values for business processes with large size.

References

- Hofacker, I., Vetschera, R.: Algorithmical approaches to business process design. *Computers & Operations Research* 28, 1253–1275 (2001)
- Tiwari, A., Vergidis, K., Majeed, B.: Evolutionary Multi-objective Optimization of Business Processes. In: *Proceedings of IEEE Congress on Evolutionary Computation 2006*, pp. 3091–3097 (2006)

- Vergidis, K., Tiwari, A.: Business Process Design and Attribute Optimization within an Evolutionary Framework. In: Proceedings of the Congress on Evolutionary Computing (CEC 2008), pp. 668–675 (2008)
- Vergidis, K., Tiwari, A., Majeed, B.: Business process improvement using multi-objective optimization. *BT Technology Journal* 24(2), 229–235 (2006)
- Vergidis, K., Tiwari, A., Majeed, B.: Composite business processes: An evolutionary multi-objective optimization approach. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC 2007), Singapore, pp. 2672–2678 (2007)
- Wang, K., Salhi, A., Fraga, E.S.: Process design optimization using embedded hybrid visualization and data analysis techniques within a genetic algorithm optimization framework. *Chemical Engineering and Processing* 43, 663–675 (2004)

Computational Intelligence in Speech and Audio Processing: Recent Advances

About Ella Hassanien, Gerald Schaefer, and Ashraf Darwish

Abstract. Computational intelligence techniques have been used for the processing of speech and audio for several years. Some of the applications in speech processing where computational intelligences are extensively used include speech recognition, speaker recognition, speech enhancement, speech coding and speech synthesis, while in audio processing, computational intelligence applications include music classification, audio classification and audio indexing and retrieval. In this paper we provide an overview of recent applications of modern computational intelligence theory in the field of speech and audio processing.

1 Introduction

Over the last few decades we have seen a new era of artificial intelligence emerging that is focussing on the principles, theoretical aspects, and design methodology of algorithms gleaned from nature. Examples include artificial neural networks inspired by mammalian neural systems, evolutionary computation inspired by natural selection in biology, simulated annealing inspired by thermodynamics principles and swarm intelligence inspired by collective behaviour of insects or micro-organisms interacting locally with their environment, causing coherent functional global patterns to emerge. Computational intelligence is a well-established paradigm with current systems having many of the characteristics of biological computers, capable of performing a variety of tasks that are difficult or impossible to do using conventional techniques.

About Ella Hassanien

Information Technology Department, Cairo University, Giza, Egypt

Gerald Schaefer

Department of Computer Science, Loughborough University, Loughborough, U.K.

Ashraf Darwish

Computer Science Department, Helwan University, Cairo, Egypt

Recent trends aim at integration of different components to take advantage of complementary features, and to develop synergistic systems leading to hybrid architectures such as neuro-fuzzy systems, evolutionary-neural networks, or rough-neural approaches for problem solving.

Computational intelligence techniques have been used for the processing of speech and audio for several years (Guido et al, 2007; Kung and Hwang, 1998). Some of the applications in speech processing where computational intelligences are extensively used include speech recognition, speaker recognition, speech enhancement, speech coding, and speech synthesis, while in audio processing, computational intelligence applications include music classification, audio classification, and audio indexing and retrieval. In this paper we provide an overview of recent advances of modern computational intelligence theory in the field of speech and audio processing.

2 Speech and Audio Processing

Speech processing comprises the study of speech signals and the processing methods of these signals. As signals are usually processed in a digital representation, speech processing can also be seen as an intersection of digital signal processing and natural language processing. Within speech processing we can categorise applications and tasks into the following: (1) Speech recognition, whose aim to analyse the linguistic content of speech signals, (2) Speaker recognition, which aims at recognising the identity of the speaker; (3) Speech coding, a specialised form of data compression which is important especially in the telecommunication area; (4) Speech synthesis to automatically generate speech, and (5) Speech enhancement, which deals with enhancing the perceptual quality of speech signal by removing noise and other factors.

Speech synthesisers have been developed for many popular languages, however designing a synthesiser for any language is largely dependant on the language structure. Text-to-speech conversion has traditionally been performed either by concatenating short samples of speech or by using rule-based systems to convert a phonetic representation of speech into an acoustic representation, which is subsequently converted into speech. Karaali et al (1996) described a system that employs a time-delay neural network to perform phonetic-to-acoustic mapping, while another neural network is used to control the timing of the generated speech. The neural network system requires less memory than a concatenation system, and was shown to perform well in comparison to commercial systems that use other technologies. It is reported that the neural network approach to speech synthesis offers the benefits of language portability, naturally sounding speech, and low storage requirements as well as providing better voice quality compared to traditional approaches. Hendessi et al (2005) presented a text-to-speech conversion system that includes a text analyser which consists of text-to-speech and phoneme-to-speech modules. A neural network is employed to extract Persian phonetics.

In addition, a new model is used as a post-processor to compensate for errors generated by the neural network. The authors also pointed out that their system can be easily adapted for other similar languages such as Arabic.

Synthesising speech from a phonetic representation using neural networks requires such networks to compute one output for each speech frame from the vocoder, a process which proves to be very expensive. [Corrigan et al \(2000\)](#) established an implementation to model the speech as a series of gestures, and generated parameters aiming to describe the transitions of the vocoder parameters during these gestures. The results of such an implementation confirmed that acceptable quality of the speech is produced when each gesture is half of a phonetic segment and the transition model is a set of cubic polynomials describing the variation of each vocoder parameter during the gesture.

Based on articulatory parameters, [Frankel et al \(2000\)](#) described a system that first estimates articulatory trajectories from speech signals. These parameters represent the basic features and phone-dependent linear dynamic models in order to perform speech recognition. Estimations of x and y coordinates of seven articulator positions in the midsagittal plane are produced every two milliseconds by a recurrent neural network, which was trained on real articulatory data. Phone recognition is performed after the resulting output of this network is then passed to a set of linear dynamic models.

Recently, the features derived from posteriors of a multilayer perceptron have been demonstrated to be very effective for automatic speech recognition. Most tandem features have relied on MLPs trained for phone classification. Based on a relatively small data set, MLPs trained for articulatory feature classification can be equally effective as pointed out by [Cetin et al \(2007\)](#). The authors provided a similar comparison using MLPs trained on a much larger data set and also explored the portability of phone- and articulatory feature-based tandem features into a different language (Mandarin) without the need to retrain the neural network. The authors reported that while phone-based features perform slightly better in the matched-language condition, they perform significantly better in the cross-language case. However, in the cross-language condition, neither approach is as effective as the tandem features extracted from an MLP trained on a relatively small amount of in-domain data. Cetin et al. also explored observation modelling schemes, that allow for greater flexibility in combining the tandem and standard features at hidden Markov model outputs.

A new approach to speech recognition using fuzzy modeling was presented by [Halavati et al \(2004\)](#). The task commences with the conversion of the speech spectrogram into a linguistic description. While phonemes are also described using these fuzzy measures, and recognition is done by fuzzy reasoning, a genetic algorithm is employed to optimise phoneme definitions so as to classify samples into correct phonemes. The method was tested over a standard speech data base.

One of the complications with speech signals is their large degree of acoustic variability. To overcome this issue, a speech recognition model using

genetic algorithms was formulated by [Bovbel and Tsishkoi \(2000\)](#). In their work, the authors performed experiments with a database of separated Belarussian words on which they achieved optimal results.

[Ding \(2007\)](#) presented FLC-MLLR, a fuzzy control mechanism for conventional maximum likelihood linear regression speaker adaptation, where the effect of MLLR adaptation is regulated according to the availability of adaptation data in a way that the advantage of MLLR adaptation can be fully used with sufficient training data, or where the consequence of poor MLLR adaptation would be restrained otherwise. The robustness of MLLR adaptation helps in particular in cases of data scarcity. The authors pointed out that their proposed mechanism is simple and of low computational complexity.

[Kostek and Czyzewski \(2001\)](#) discussed some drawbacks of the hearing-aid fitting process, where an audiologist executes tests on the wearer of the hearing aid which is then adjusted based on the results of the test. The aim is to make the device work as best as it can for each individual. The presented automatic system for fitting hearing aids employs fuzzy logic and makes choices for adjusting the hearing aids settings by analysing the patient's responses to a set of questions.

Speech training can be made available to patients without continuous assistance required from speech therapists, due to the increasing access to multimedia computers. Thus, screening of speech fluency can also be performed, providing immediate information to speech disorder patients and those who have problems with understanding speech. [Czyzewski et al \(2002\)](#) introduced such a speech therapy training algorithm consisting of diagnostic tools and rehabilitation devices connected with it. The first function that the system performs is data acquisition which happens when information about the patient's medical history is collected. This is done through electronic questionnaires. The following function is the analysis of the speech signal articulated by the patient when prompted by the computer followed by multimedia tests carried out in order to assess the subject's ability to understand speech. The results of the electronic questionnaire, the patient's voice and patient's reactions are then automatically analysed. Finally, the system automatically diagnoses possible speech disorders and their strengths.

Analysing stuttering can be performed more objectively through the automatic detection of stop-gaps, syllable repetitions and vowel prolongations. Compared to this, the alternative would be based on subjective evaluations of speech fluency which may be dependent on a subjective evaluation method. Automatic detection of intervocalic intervals, stop-gaps, voice onset time and vowel durations may depend on the speaker and therefore rules derived for a single speaker might be unreliable when trying to generalise them. This implies that strong generalisation capabilities of learning algorithms, could be applied to solve the problem. Still, such a system requires some parameters, which characterise the distinctive features in a subject's speech patterns. Moreover, an appropriate selection of the parameters and feature vectors while learning may improve the performance of an automatic

detection system. [Czyzewski et al \(2003\)](#) presented an approach to automatic recognition of stuttered speech in normal and frequency altered feedback speech. They proposed several methods of analysing stuttered speech and describe how to establish those parameters. They also reported results of experiments on automatic detection of speech disorder events based on both rough sets and artificial neural networks.

[Czyzewski and Szczerba \(2002\)](#) introduced an approach for pitch estimation enhancement which addresses the disadvantages of standard pitch estimation algorithms. The presented algorithm is based on both pitch estimation and pitch prediction in terms of signal processing and musical knowledge modelling. First, the signal is partitioned into segments which are roughly analogous to consecutive notes. Then, for each segment an autocorrelation function is calculated, and the autocorrelation function values altered using a music predictor based on artificial neural networks.

A hybrid decision fusion model and robust continuous digital speech recognition using support vector machine and duration distribution based hidden Markov model was proposed by [Liu et al \(2007\)](#) who investigated probability output combination of support vector machines and Gaussian mixture models, and embedding the fusion probability as similarity into the phone state level decision space of the duration distribution based hidden Markov model (DDBHMM) speech recognition system. The performance of FSVM and FSVM/DDBHMM were compared on a continuous Mandarin digital speech corpus in four noisy environments.

Methods for the identification of the direction of the incoming acoustical signal in the presence of noise and reverberation were investigated by [Czyzewski \(2003\)](#). As the problem is non-deterministic, two learning algorithms, namely neural networks and rough sets, were employed. From these, two sets of parameters are formulated in order to discern the target source from unwanted sound source positions.

[Zwan et al \(2007\)](#) presented an automatic singing voice recognition algorithm which is based on neural networks and rough sets. A database containing singers' sample recordings is built, and parameters are extracted from recorded voices of trained and untrained singers of various voice types. These parameters are especially designed for the analysis of singing voices. Artificial neural networks and rough sets are then used for automatic voice type/voice quality classification.

[Selouani and O'Shaughnessy \(2003\)](#) proposed a hybrid enhancement noise reduction approach in the cepstral domain in order to obtain robust parameters. Principal component analysis is used in the Mel frequency domain coupled with a genetic algorithm. The authors demonstrated that the enhanced parameters increased the recognition rate for highly interfering noise environments. The proposed hybrid enhancement technique outperformed the conventional recognition process in severe interfering car noise environments for a wide range of signal-to-noise ratios varying from 16 dB to -4 dB, when included in the front-end of an HTK-based speech recognition system. The

authors also showed the effectiveness of their approach in recognising speech subject to telephone channel degradations.

3 Speech Emotion Recognition

Speech emotion recognition is becoming more and more important in various application fields including health care and education. However, relatively little work has been conducted on performing speech emotion recognition using computational intelligence methods. Of central importance is the selection of an appropriate feature set for subsequent classification, yet the derivation of optimal feature sets is a hard problem and many existing techniques are of high computational complexity. To address this issue, [Zhou et al \(2006\)](#) presented an approach based on rough set theory and support vector machines for speech emotion recognition. They showed that their method is able to reduce the computational complexity while maintaining a high recognition rate. [Schuller et al \(2006\)](#) used evolutionary programming to provide a faster approach for speech emotion feature generation and showed improved results on two public databases.

[Fellenz et al \(2000\)](#) also addressed the problem of identifying appropriate features for emotion recognition, based on speech and face image sequences. They employed a combination of neural networks and fuzzy logic to develop a hybrid classification system that is able to detect presence of emotional expressions of the speaker.

[Buscicchio et al \(2006\)](#) focussed on extracting vowel information from an input speech signal and employed a spiking neural network (SNN) to classify the extracted information. The speech signal is segmented into vowel parts, and a set of salient features related to the Mel frequency spectrum, are derived for each vowel part. A SNN is then trained with this features a learns to differentiate between five different emotion classes.

4 Audio-Visual Speech Recognition

Human listeners are able to exploit visual cues, such as lip and tongue movements, in order to enhance our ability of speech understanding, especially in noisy environments. Audio visual speech recognition (AVSR) ([Nakamura, 2002](#)) try to mimic this ability by utilising additional sensor information, in particular video data of the speaker, to aid and improve speech recognition, often using lip reading approaches. Computational intelligence techniques having been shown to play an important role in this research direction with a number of CI-based AVSR methods having been proposed in the literature.

Clearly, the performance of an AVSR system heavily relies on a good and robust set of visual features. [Lim et al \(2004\)](#) addressed the problem of mouth tracking and employed radial basis function neural networks to improve mouth tracking performance. They used a modified extended Kalman

filter to adjust the parameters of the neural network and show that their method affords good recognition performance. [Sadeghi and Yaghmaei \(2006\)](#) also employed neural networks. They extracted lip features from the visual data and used these as inputs for a neural network. The network is then trained to recognise the main Farsi vowels.

[Lewis and Powers \(2002\)](#) discussed how to improve the performance of speech recognition systems by using information from both auditory and visual signals. Using knowledge from psycholinguistics, a late integration network was developed to fuse sources. The authors examined how useful a proposed mouth region extraction technique is in combination with several integration architectures and demonstrated that vision can indeed be used to assist speech recognition when used in a linguistically guided fashion.

[Bugatti et al \(2002\)](#) addressed the problem of audio classification in speech and music, and compared a statistical approach to one that is based on neural networks. The statistical technique they employed is based on zero crossings and a Bayesian classifier. Although simple from a computational point of view, it provides good results for pure music or speech. However, performance degrades in cases where speech and music are superimposed, or where strong rhythmic components are present. To overcome these problems, a method based on an extended feature set and a neural network classifier was introduced which in turn is shown to lead to improved recognition performance.

[Faraj and Bigun \(2007\)](#) described lip-motion as the distribution of apparent velocities in the movement of brightness patterns in the visual data. This provides a more robust approach to feature extraction as no explicit lip contours are estimated. The presented approach is then employed in a person authentication system which is based on lip-movements and speech. Experimental results show a high recognition rate of 98%.

[Meng and Zhang \(2003\)](#) described a method of visual speech feature area localisation. They used a simplified human skin color model to segment input images and to estimate the location of the face. Then, a localisation method that uses a combination of support vector machines and distance of likelihood in feature space derived from kernel principal component analysis, is employed. Results, based on a Chinese audio-visual speech database, show that the proposed method outperforms traditional linear ones.

5 Conclusions

In this paper we have reviewed recent advances in the application of computational intelligence techniques for audio and speech processing. Clearly, audio and speech are often combined with other media to form multimedia. During the last decade, multimedia processing has emerged as an important technology to generate content based on images, video, audio, graphics, and text. Furthermore, recent developments in high-definition multimedia content and

interactive television will generate huge volumes of data leading to important computing problems connected with the creation, processing and management of multimedia content. Multimedia computing is a challenging domain for several reasons: it requires both high computation rates and memory bandwidth, it is a multi-rate computing problem, and it requires low-cost implementations for high-volume markets. Both multimedia processing in general, and audio and speech processing in particular are challenging areas for computational intelligence to play crucial roles in resolving problems and providing solutions, and we can expect many more interesting developments in this area to come.

References

- Bovbel, E., Tsishkou, D.: Belarussian speech recognition using genetic algorithms. In: Sojka, P., Kopeček, I., Pala, K. (eds.) TSD 2000. LNCS (LNAI), vol. 1902, pp. 185–204. Springer, Heidelberg (2000)
- Bugatti, A., Flammini, A., Migliorati, P.: Audio classification in speech and music: a comparison between a statistical and a neural approach. *EURASIP Journal on Applied Signal Processing* 2002 (4), 372–378 (2002)
- Buscicchio, C., Grecki, P., Caponetti, L.: Speech emotion recognition using spiking neural networks. In: Esposito, F., Raś, Z.W., Malerba, D., Semeraro, G. (eds.) ISMIS 2006. LNCS (LNAI), vol. 4203, pp. 38–46. Springer, Heidelberg (2006)
- Cetin, O., Kantor, A., King, S., Bartels, C., Magimai-Doss, M., Frankel, J., Livescu, K.: An articulatory feature-based tandem approach and factored observation modeling. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, pp. 645–648 (2007)
- Corrigan, G., Massey, N., Schnurr, O.: Transition-based speech synthesis using neural networks. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 945–948 (2000)
- Czyzewski, A.: Automatic identification of sound source position employing neural networks and rough sets. *Pattern Recognition Letters* 24, 921–933 (2003)
- Czyzewski, A., Szczerba, M.: Pitch estimation enhancement employing neural network-based music prediction. In: *IASTED Intern. Conference, Artificial Intelligence and Soft Computing*, pp. 413–418 (2002)
- Czyzewski, A.B.K., Skarzynski, H.: Diagnostic system for speech articulation and speech understanding. In: *Meeting of the Acoustical Society of America* (2002)
- Czyzewski, A., Kaczmarek, A., Kostek, B.: Intelligent processing of stuttered speech. *Journal of Intelligent Information Systems* 21(2), 143–171 (2003)
- Ding, I.J.: Incremental MLLR speaker adaptation by fuzzy logic control. *Pattern Recognition* 40(11), 3110–3119 (2007)
- Faraj, M., Bigun, J.: Audio-visual person authentication using lip-motion from orientation maps. *Pattern Recognition Letters* 28(11), 1368–1382 (2007)
- Fellenz, W., Taylor, J., Cowie, R., Douglas-Cowie, E., Piat, F., Kollias, C., Orovas, S., Apolloni, B.: On emotion recognition of faces and of speech using neural networks, fuzzy logic and the ASSESS system. In: *IEEE-INNS-ENNS International Joint Conference on Neural Networks*, vol. 2, pp. 93–98 (2000)

- Frankel, J., Richmond, K., King, S., Taylor, P.: An automatic speech recognition system using neural networks and linear dynamic models to recover and model articulatory traces. In: Proc. ICSLP, vol. 4, pp. 254–257 (2000)
- Guido, R., Pereira, J., Slaets, J.: Advances on pattern recognition for speech and audio processing. *Pattern Recognition Letters* 28(11), 1283–1284 (2007)
- Halavati, R., Shouraki, S., Eshraghi, M., Alemzadeh, M., Ziaie, P.: A novel fuzzy approach to speech recognition. In: International Conference on Hybrid Intelligent Systems, pp. 340–345 (2004)
- Hendessi, F., Ghayoori, A., Gulliver, T.A.: A speech synthesizer for Persian text using a neural network with a smooth ergodic HMM. *ACM Transactions on Asian Language Information Processing* 4(1), 38–52 (2005)
- Karaali, O., Corrigan, G., Gerson, I.: Speech synthesis with neural networks. In: World Congress on Neural Networks, pp. 45–50 (1996)
- Kostek, B., Czyzewski, A.: Employing fuzzy logic and noisy speech for automatic fitting of hearing aid. In: Meeting of the Acoustical Society of America (2001)
- Kung, S.Y., Hwang, J.N.: Neural networks for intelligent multimedia processing. *Proceedings of the IEEE* 86(6), 1244–1272 (1998)
- Lewis, T., Powers, D.M.W.: Audio-visual speech recognition using red exclusion and neural networks. In: Australasian conference on Computer science, vol. 4, pp. 149–156 (2002)
- Lim, E., Seng, K., Tse, K.: RBF neural network mouth tracking for audio-visual speech recognition system. In: IEEE Region 10 Conference TENCN, pp. 84–87 (2004)
- Liu, J., Wang, Z., Xiao, X.: A hybrid SVM/DDBHMM decision fusion modeling for robust continuous digital speech recognition. *Pattern Recognition Letter* 28(8), 912–920 (2007)
- Meng, S., Zhang, Y.: A method of visual speech feature area localization. In: International Conference on Neural Networks and Signal Processing, vol. 2, pp. 1173–1176 (2003)
- Nakamura, S.: Statistical multimodal integration for audio-visual speech processing. *IEEE Transactions on Neural Networks* 13(4), 854–866 (2002)
- Sadeghi, V., Yaghmaie, K.: Vowel recognition using neural networks. *International Journal of Computer Science and Network Security* 6(12), 154–158 (2006)
- Schuller, B., Reiter, S., Rigoll, G.: Evolutionary feature generation in speech emotion. In: IEEE International Conference on Recognition Multimedia, pp. 5–8 (2006)
- Selouani, S.A., O’Shaughnessy, D.: On the use of evolutionary algorithms to improve the robustness of continuous speech recognition systems in adverse conditions. *EURASIP Journal on Applied Signal Processing* 8, 814–823 (2003)
- Zhou, J., Wang, G., Yang, Y., Chen, P.: Speech emotion recognition based on rough set and SVM. In: 5th IEEE International Conference on Cognitive Informatics, vol. 1, pp. 53–61 (2006)
- Zwan, P., Szczuko, P., Kostek, B., Czyzewski, C.: Automatic singing voice recognition employing neural networks and rough sets. In: Kryszkiewicz, M., Peters, J.F., Rybiński, H., Skowron, A. (eds.) RSEISP 2007. LNCS (LNAI), vol. 4585, pp. 793–802. Springer, Heidelberg (2007)

Author Index

- Abonyi, János 31, 117
Aguilar, Luis T. 229, 245
Ahamadi, Malidi 91
Akbarzadeh Totonchi, Mohammad
Reza 175, 263
Aliyari Shoorehdeli, Mahdi 41
AlQaheri, Hameed 187
Arshed, Jawad Usman 205
Asadpour, Vahid 175
- Badie, Kambiz 127
Balasubramaniam, Sheela Rani 157
Bankó, Zoltán 117
Barai, S.V. 61
Borges, Patrick 53
- Caggiani, Leonardo 255
Campos, Mauro 53
Castillo, Oscar 217, 229, 245
Cazarez-Castro, Nohe R. 229
Covas, José 3
- Darwish, Ashraf 303
de Araújo, Leonardo Carneiro 283
de Carvalho, Bernardo Penna Resende
283
Duarte, João 109
- Gaspar-Cunha, António 3, 109
- Hassanien, Aboul Ella 187, 303
Hiro, Takafumi 139
- Iannucci, Giuseppe 255
Ichiyama, Shohei 139
- Iqbal, Muhammad 205
Iswandy, Kuncup 149
- Kamkar, Iman 165, 263
Koga, Takanori 139
Koh, Andrew 71
König, Andreas 149
Krohling, Renato A. 53
Kuo, Po-Hsien 271
- Li, Ying 271
- Madani, Sajjad Ahmad 205
Martinez, Ricardo 245
Marusak, Piotr Marek 11
Matsuzaki, Masunori 139
Melin, Patricia 217
Mendes, Fernando 109
Mendyk, Aleksander 91
Milanović, Marija 81
Moghimi, Ali 175
Mohebbi, Javad 165
Montiel-Ross, Oscar 217
- Nakashima, Tomoharu 101
Narayanasamy, Manoharan 157
Noruzi Nashalji, Mostafa 41
- Ottomanelli, Michele 255
- Perumal, Kalyasundaram 157
Polak, Sebastian 91
Poostchi, Mahdieh 165, 263
- Qiao, Fengxiang 271

- Raj, Baldev 157
Rao, Lakshmesha 149
Rathinasabapathy, Ramadevi 157
Rodriguez, Antonio 245
- Sadeghzadeh, Mehdi 127
Sarfraz, Muhammad 205
Sassanelli, Domenico 255
Schaefer, Gerald 101, 187, 197, 303
Sepúlveda, Roberto 217
Srikanth, D. 61
Stütze, Thomas 3
Suetake, Noriaki 139
- Tait, Roger 197
Teixeira, Cristina 3
Teshnehlab, Mohammad 41, 127
- Tiwari, Ashutosh 293
Turner, Chris 293
- Uchino, Eiji 139
- Vasudevan, Prakash 157
Vatankhah, Maryam 175
Vergidis, Kostas 293
Vieira, Armando 109
- Wiśniowska, Barbara 91
Wu, Hao 237
Wu, Zhengping 237
- Yu, Lei 271
- Zhu, Qing 271
Zidek, Jan 21

Subject Index

- access control 237
- anisotropic diffusion 139
- ant colony optimization 127
- artificial neural networks 91
- audio-visual speech recognition 303
- audio processing 303

- backlash 229
- bankruptcy prediction 109
- bare bones particle swarm optimization 53
- bicycle 271
- bioinspired 21
- blackboard 199
- business processes 293

- cavitation phenomenon 157
- cellular genetic algorithm 263
- classification 127
- collaborative social network 237
- comb filter 165
- combinatorial optimization 81
- computational intelligence 303
- constraint handling 61
- contrast enhancement 187
- crossover 263
- cutoff 21

- data mining 31, 109, 127
- decision tree 31, 283
- differential evolution 71
- distributed processing 197
- DTW 117

- embedded Karnik-Mendel type reducer 217
- embedded type-2 fuzzy system 217
- equilibrium problems with equilibrium constraints 71
- evolutionary algorithms 3, 109
- evolutionary computation 81, 245

- facility planning and design 81
- fault detection 41
- feature selection 127
- feed forward neural network 157
- Fourier descriptors 205
- FPGA type-2 FIS 217
- fuzzy c-means 187, 271
- fuzzy classification 101
- fuzzy classifier 31
- fuzzy control 11
- fuzzy inference 139, 283
- fuzzy logic 187, 237, 255
- fuzzy logic controllers 245
- fuzzy Lyapunov synthesis 229
- fuzzy rule base 101
- fuzzy systems 11

- gap acceptance 255
- gene expression analysis 101
- genetic algorithms 41, 81, 101, 127, 149, 245, 293
- genetic programming 283

- harmony search 61
- hERG potassium channel 91
- high speed type-2 fuzzy system 217
- hybrid classification 101

- image segmentation 187
- intelligent agents 199
- intravascular ultrasound (IVUS) image 139
- least squares support vector machine 283
- localization 149
- long QT syndrome 91
- medical image processing 139
- mixed Weibull distribution 53
- modeling 21
- multi-objective algorithm 283
- multi-objective optimization 3, 109, 293
- multimedia computing 197
- music visualization 165
- mutation 263
- Nash dominance 71
- Nash equilibrium 71
- neural classifier 41
- neural networks 255
- noise 205
- non-linear feature 175
- nonlinear control 11
- object recognition 205
- optimisation framework 293
- optimizing control 11
- P2P technology 237
- PAES 293
- parallel processing 198
- Pareto local search 3
- particle swarm optimization 245
- pattern recognition 271
- pedestrian 255
- pedestrian crosswalk 271
- piecewise linear approximation 117
- plaque boundary extraction 139
- point cloud 21
- polymer extrusion 3
- potassium channel inhibition prediction 91
- predictive control 11
- principal component analysis 41
- prostate cancer 187
- rapid eye movement 175
- recursive feature elimination 283
- relax time 165
- reliability 53
- REM 175
- resilient back propagation algorithm 157
- resource allocation 237
- rhythm components 165
- ribosome 21
- right turn vehicle 271
- Sammon's mapping 149
- segmentation 117
- sensor networks 149
- simulated evolution 205
- sleep stage 175
- spectral feature 175
- speech emotion recognition 303
- speech processing 303
- structural optimization 61
- SVM 175
- swarm optimization 149
- Tennessee-Eastman process 41
- time windows 263
- TRUS 187
- truss 61
- type-2 fuzzy control 229
- type-II fuzzy sets 187
- ultrasound imaging 187
- unsignalized crossings 255
- vehicle routing 263
- video streaming 237
- weight minimization 61