

LNAI 6881

Andreas König Andreas Dengel
Knut Hinkelmann Koichi Kise
Robert J. Howlett Lakhmi C. Jain (Eds.)

Knowledge-Based and Intelligent Information and Engineering Systems

15th International Conference, KES 2011
Kaiserslautern, Germany, September 2011
Proceedings, Part I

1 Part I



Springer

LNAI Series Editors

Randy Goebel

University of Alberta, Edmonton, Canada

Yuzuru Tanaka

Hokkaido University, Sapporo, Japan

Wolfgang Wahlster

DFKI and Saarland University, Saarbrücken, Germany

LNAI Founding Series Editor

Joerg Siekmann

DFKI and Saarland University, Saarbrücken, Germany

Andreas König Andreas Dengel
Knut Hinkelmann Koichi Kise
Robert J. Howlett Lakhmi C. Jain (Eds.)

Knowledge-Based and Intelligent Information and Engineering Systems

15th International Conference, KES 2011
Kaiserslautern, Germany, September 12-14, 2011
Proceedings, Part I

Series Editors

Randy Goebel, University of Alberta, Edmonton, Canada
Jörg Siekmann, University of Saarland, Saarbrücken, Germany
Germany

Volume Editors

Andreas König
University of Kaiserslautern, Germany
E-mail: koenig@eit.uni-kl.de

Andreas Dengel
DFKI and University of Kaiserslautern, Germany
E-mail: andreas.dengel@dfki.de

Knut Hinkelmann
University of Applied Sciences Northwestern Switzerland, Olten, Switzerland
E-mail: knut.hinkelmann@fhnw.ch

Koichi Kise
Osaka Prefecture University, Osaka, Japan
E-mail: kise@cs.osakafu-u.ac.jp

Robert J. Howlett
KES International, Shoreham-by-sea, UK
E-mail: rjhowlett@kesinternational.org

Lakhmi C. Jain
University of South Australia, Adelaide, SA, Australia
E-mail: lakhmi.jain@unisa.edu.au

ISSN 0302-9743

e-ISSN 1611-3349

ISBN 978-3-642-23850-5

e-ISBN 978-3-642-23851-2

DOI 10.1007/978-3-642-23851-2

Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011935629

CR Subject Classification (1998): I.2, H.4, H.3, I.4-5, H.5, C.2, H.2.8

LNCS Sublibrary: SL 7 – Artificial Intelligence

© Springer-Verlag Berlin Heidelberg 2011

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, re-use of illustrations, recitation, broadcasting, reproduction on microfilms 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 to 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.

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The 15th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems was held during September 12–14, 2011 in Kaiserslautern, Germany. The conference was hosted by the University of Kaiserslautern and the German Research Center for Artificial Intelligence (DFKI) GmbH, Germany, and KES International.

KES 2011 provided a scientific forum for the presentation of the results of high-quality international research including recent results of large-scale projects, new exciting techniques, and models, as well as innovative solutions in challenging application fields. The conference attracted contributions from 32 countries and 5 continents: Australia, Canada, China, Colombia, Croatia, Czech Republic, Finland, France, Germany, Greece, Indonesia, Iran, Italy, Japan, Jordan, Korea, Latvia, Malaysia, Mexico, Norway, Poland, Romania, Russia, Spain, Sweden, Switzerland, Taiwan, Thailand, Tunisia, Turkey, UK, and USA.

The conference consisted of 6 keynote talks, 9 general tracks and 25 invited sessions and workshops, on the advance and application of knowledge-based and intelligent systems and related areas. The distinguished keynote speakers were:

Ansgar Bernardi

German Research Center for Artificial Intelligence, Kaiserslautern, Germany

“Growing Together: Opening the Way for Comprehensive Public–Private Knowledge Management”

Knut Manske

Vice President SAP Research, SAP AG, Darmstadt, Germany

“Future Urban Management: Towards Best Managed Cities”

Nikhil R. Pal

Indian Statistical Institute, Calcutta, India

“Selection of Useful Sensors/Features with Controlled Redundancy Using Neural Networks”

Peter Schütt

Leader Software Strategy & Knowledge Management, Executive Engagement Manager, IBM Software Group Germany

“Knowledge Sharing in Enterprise Networks”

Ulrich Reimer

Institute for Information and Process Management University of Applied Sciences St. Gallen, Switzerland

“(Meta-) Modeling of Process-Oriented Information Systems”

Keiji Yamada

General Research Manager, C&C innovation Laboratories, NEC Corporation
Professor, Nara Institute of Science and Technology

“Symbiotic System as a New Social Infrastructure Based on Intelligent Interaction Among the Society, Human Beings, and Information Systems”

Overall 244 oral presentations, complemented by focused lab tours at the organizing institutions, provided excellent opportunities for the presentation of intriguing new research results and vivid discussion on these, paving the way to efficient knowledge transfer and the incubation of new ideas and concepts.

As in the previous years, extended versions of selected papers were considered for publication in follow-up journal publications.

We would like to acknowledge the contribution of the Track Chairs, Invited Sessions Chairs, all members of the Program Committee and external reviewers for coordinating and monitoring the review process. We are grateful to the editorial team of Springer led by Alfred Hofmann. Our sincere gratitude goes to all participants and the authors of the submitted papers.

September 2011

Andreas Dengel
Andreas König
Koichi Kise
Knut Hinkelmann
Robert Howlett
Lakhmi Jain

Organization

KES 2011 was hosted and organized by the Chair's Knowledge-Based Systems, Computer Science department, and Integrated Sensor Systems, Electrical and Computer Engineering department at the University of Kaiserslautern, the German Research Center for Artificial Intelligence (DFKI) GmbH, Germany, and KES International. The conference was held at the University of Kaiserslautern, September 12–14, 2011.

Executive Committee

General Co-chairs

Andreas Dengel	University of Kaiserslautern and DFKI GmbH, Germany
Andreas König	University of Kaiserslautern, Germany
Lakhmi Jain	University of South Australia, Australia

Executive Chair

Robert Howlett	Bournemouth University, UK
----------------	----------------------------

Program Co-chairs

Knut Hinkelmann	University of Applied Sciences Northwestern Switzerland, Switzerland
Koichi Kise	Osaka Prefecture University, Japan

Organizing Committee Chair

Stefan Zinsmeister	DFKI GmbH, Germany
--------------------	--------------------

Organizing Committee

KES Operations Manager

Peter Cushion	KES International, UK
---------------	-----------------------

KES Systems Support

Shaun Lee	KES International, UK
-----------	-----------------------

ISE Support Staff

Abhaya Chandra Kammara	University of Kaiserslautern, Germany
Shubhmoy Kumar	University of Kaiserslautern, Germany

Track Chairs

Bruno Apolloni	University of Milan, Italy
Floriana Esposito	University of Bari, Italy
Anne Håkansson	Stockholm University, Sweden
Ron Hartung	Franklyn University, USA
Honghai Liu	University of Portsmouth, UK
Heiko Maus	DFKI GmbH, Germany
Ngoc Thanh Nguyen	Wroclaw University of Technology, Poland
Andreas Nuernberger	University of Magdeburg, Germany
Tuan Pham	University of New South Wales, Australia
Toyohide Watanabe	Nagoya University, Japan

Invited Session Chairs

The Second International Workshop on Natural Language Visualization

Minhua Ma	The Glasgow School of Art, UK
Bob Coyne	Columbia University, USA

Workshop on Seamless Integration of Semantic Technologies in Computer-Supported Office Work (SISTCOW)

Oleg Rostanin	DFKI GmbH, Germany
Simon Scerri	University of Ireland, Galway, Ireland
Benedikt Schmidt	SAP Research, Germany

Innovations in Chance Discovery

Akinori Abe	University of Tokyo, Japan
Yukio Ohsawa	The University of Tokyo, Japan

Computational Intelligence Methods to Benefit Society

Valentina Balas	Aurel Vlaicu University of Arad, Romania
Lakhmi C. Jain	University of South Australia, Australia

Knowledge-Based Interface Systems (I)

Yuji Iwahori	Chubu University, Japan
Naohiro Ishii	Aichi Institute of Technology, Japan

Advances in Theory and Application of Hybrid Intelligent Systems

Lakhmi C. Jain	University of South Australia, Australia
CP Lim	Universiti Sains Malaysia, Malaysia

Recent Trends in Knowledge Engineering, Smart Systems and Their Applications

Cesar Sanin University of Newcastle, Australia
Carlos Toro VICOMTech, Spain

Data Mining and Service Science for Innovation

Katsutoshi Yada Kansai University, Japan

Methods and Techniques of Artificial and Computational Intelligence in Economics, Finance and Decision Making

Marina Resta DIEM sezione di Matematica Finanziaria, Italy

Human-Oriented Learning Technology and Learning Support Environment

Toyohide Watanabe Nagoya University, Japan
Tomoko Kojiri Nagoya University, Japan

Human Activity Support in Knowledge Society

Toyohide Watanabe Nagoya University, Japan
Takeshi Ushijima Kyushu University, Japan

Design of Social Intelligence and Creativity Environment

Toyohide Watanabe Nagoya University, Japan
Naoto Mukai Tokyo University of Science, Japan

Knowledge Engineering Applications in Process Systems and Plant Operations

Kazuhiro Takeda Shizuoka University, Japan
Takashi Hamaguchi Nagoya Institute of Technology, Japan
Tetsuo Fuchino Tokyo Institute of Technology, Japan

Knowledge - Based Interface Systems (II)

Yoshinori Adachi Chubu University, Japan
Nobuhiro Inuzuka Nagoya Institute of Technology, Japan

Emergent Intelligent Technologies in Multimedia Information Processing (IMIP)

Giovanna Castellano University of Bari, Italy
Maria Alessandra Torsello University of Bari, Italy

Time Series Prediction Based on Fuzzy and Neural Networks

Minvydas Ragulskis

Kaunas University of Technology, Lithuania

Management Technologies from the Perspective of Kansei Engineering and Emotion

Junzo Watada

Waseda University, Japan

Hisao Shiizuka

Kogakuin University, Japan

Taki Kanda

Bunri University of Hospitality, Japan

Knowledge-Based Systems for e-Business

Kazuhiko Tsuda

University of Tsukuba, Japan

Nubuo Suzuki

KDDI Corporation, Japan

Reasoning Based Intelligent Systems (RIS)

Kazumi Nakamatsu

University of Hyogo, Japan

Jair Minoro Abe

University of Sao Paulo, Brazil

Skill Acquisition and Ubiquitous Human–Computer Interaction

Hirokazu Taki

Wakayama University, Japan

Masato Soga

Wakayama University, Japan

International Session on Sustainable Information Systems

Anne Håkansson

KTH, Sweden

Jason J. Jung

Yeungnam University , Korea

Costin Badica

University of Craiova, Romania

Intelligent Network and Service

Jun Munemori

Wakayama University, Japan

Takaya Yuizono

Japan Advanced Institute Science and
Technology, Japan*Advances in Theory and Application of Multi-Agent Systems*

Bala M. Balachandran

University of Canberra, Australia

Dharmendra Sharma

University of Canberra, Australia

Advanced Design Techniques for Adaptive Hardware and Systems

Sorin Hintea	Technical University of Cluj-Napoca, Romania
Hernando Fernández-Canque	Glasgow Caledonian University, UK
Gabriel Oltean	Technical University of Cluj-Napoca, Romania

Advanced Knowledge-Based Systems

Alfredo Cuzzocrea	ICAR-CNR, University of Calabria, Italy
-------------------	---

Computational Intelligence for Fault Diagnosis and Prognosis

Beatrice Lazzerini	University of Pisa, Italy
Marco Cococcioni	University of Pisa, Italy
Sara Lioba Volpi	University of Pisa, Italy

Multiple Classifiers and Hybrid Learning Paradigms

Edmondo Trentin	University of Siena, Italy
Friedhelm Schwenker	University of Ulm, Germany

Soft Computing Techniques and Their Intelligent Utilizations

Norio Baba	Osaka Kyoiku University, Japan
Kunihiro Yamada	Tokai University, Japan

Document Analysis and Knowledge Science

Seiichi Uchida	Kyushu University, Japan
Marcus Liwicki	DFKI GmbH, Germany
Koichi Kise	Osaka Prefecture University, Japan

Model-Based Computing for Innovative Engineering

Klaus Schneider	University of Kaiserslautern, Germany
Norbert Wehn	University of Kaiserslautern, Germany

Immunity-Based Systems

Yoshiteru Ishida	Toyohashi University of Technology, Japan
Andreas König	University of Kaiserslautern, Germany

Program Committee

Akinori Abe	University of Tokyo, Japan
Jair Minoro Abe	University of Sao Paulo, Brazil
Canicious Abeynayake	DSTO, Australia
Yoshinori Adachi	Chubu University, Japan

Benjamin Adrian	German Research Center for Artificial Intelligence (DFKI), Germany
Plamen Angelov	Lancaster University, UK
Ahmad Taher Azar	Modern Science and Arts University (MSA), Egypt
Norio Baba	Osaka Kyoiku University, Japan
Costin Badica	University of Craiova , Romania
Bala Balachandran	University of Canberra, Australia
Valentina Balas	Aurel Vlaicu University of Arad, Romania
Vivek Bannore	University of South Australia, Australia
Adrian S. Barb	Penn State University, USA
Ansgar Bernardi	German Research Center for Artificial Intelligence (DFKI), Germany
Monica Bianchini	University of Siena, Italy
Isabelle Bichindaritz	University of Washington, USA
Veselka Boeva	Technical University of Sofia, Bulgaria
Christopher Buckingham	Aston University, UK
Giovanna Castellano	University of Bari, Italy
Barbara Catania	Università degli Studi di Genova, Italy
Michele Ceccarelli	University of Sannio, Italy
Javaan Chahl	DSTO, Australia
Stephan Chalup	The University of Newcastle, Australia
Chien-Fu Cheng	Tamkang University, Taiwan
Kai Cheng	Brunel University, UK
Benny Cheung	Honk Kong Polytechnic University, Hong Kong
Marco Cococcioni	University of Pisa, Italy
Bob Coyne	Columbia University, USA
Paolo Crippa	Università Politecnica delle Marche, Italy
Mary (Missy) Cummings	Massachusetts Institute of Technology, USA
Alfredo Cuzzocrea	ICAR-CNR & University of Calabria , Italy
Ernesto Damiani	Università degli Studi di Milano, Italy
Stamatia Dasiopoulou	Informatics and Telematics Institute, Greece
Martine De Cock	University of Washington Tacoma, USA
Philippe De Wilde	Heriot-Watt University, UK
Argyris Dentsoras	University of Patras, Greece
Liya Ding	Macau University of Science and Technology, Hong Kong
Richard J. Duro	Universidade da Coruña, Spain
Schahram Dustdar	Vienna University of Technology, Austria
Isao Echizen	National Institute of Informatics, Japan
Tapio Elomaa	Tampere University of Technology, Finland
Hernando Fernandez-Canque	Glasgow Caledonian University, UK
Ana Fernandez-Vilas	University of Vigo, Spain
Arthur Filippidis	DSTO, Australia
Tetsuo Fuchino	Tokyo Institute of Technology, Japan

Junbin Charles Gao	Sturt University, Australia
Petia Georgieva	University of Aveiro, Portugal
Daniela Godoy	UNICEN University, Argentina
Bernard Grabot	LGP-ENIT, France
Manuel Graña Romay	Universidad del País Vasco, Spain
Christos Grecos	University of West Scotland, UK
Anne Hakansson	KTH, Sweden
Takashi Hamaguchi	Nagoya Institute of Technology, Japan
Alex Hariz	University of South Australia, Australia
Mohamed Hassan	Cairo University, Egypt
Richard Hill	University of Derby, UK
Sorin Hintea	Technical University of Cluj-Napoca, Romania
Dawn Holmes	University of California, USA
Katsuhiro Honda	Osaka Prefecture University, Japan
Tzung-Pei Hong	National University of Kaohsiung, Taiwan
Eyke Hullermeier	Philipps-Universität Marburg, Germany
Nikhil Ichalkaranje	University of Mumbai, India
Nobuhiro Inuzuka	Nagoya Institute of Technology, Japan
Naohiro Ishii	Aichi Institute of Technology, Japan
Takayuki Ito	Massachusetts Institute of Technology, USA
Yuji Iwahori	Chubu University, Japan
Norbert Jastroch	MET Communications GmbH, Germany
Richard Jensen	Aberystwyth University, UK
Andrew Jones	Cardiff University, UK
Jason J. Jung	Yeungnam University, Korea
Taki Kanda	Bunri University of Hospitality, Japan
Anastasia Kastania	Athens University of Economics and Business, Greece
Hideki Katagiri	Hiroshima University, Japan
Koichi Kise	Osaka Prefecture University, Japan
In-Young Ko	KAIST, Korea
Vassilis S. Kodogiannis	University of Westminster, UK
Tomoko Kojiri	Nagoya University, Japan
Amit Konar	Jadavpur University, India
Ivan Koychev	University of Sofia, Bulgaria
Halina Kwasnicka	Wroclaw University of Technology, Poland
C.K. Kwong	The Hong Kong Polytechnic University, Hong Kong
Beatrice Lazzerini	University of Pisa, Italy
Dah-Jye Lee	Brigham Young University, USA
CP Lim	Universiti Sains Malaysia, Malaysia
Tsung-Chih Lin	Feng-Chia University, Taiwan
James Liu	The Hong Kong Polytechnic University, Hong Kong
Lei Liu	Beijing University of Technology, China

Marcus Liwicki	German Research Center for Artificial Intelligence (DFKI), Germany
Ignac Lovrek	University of Zagreb, Croatia
Jie Lu	University of Technology, Sydney, Australia
Minhua Eunice Ma	University of Derby, UK
Ilias Maglogiannis	University of Central Greece, Greece
Nadia Magnenat-Thalmann	University of Geneva, Switzerland
Dario Malchiodi	Università degli Studi di Milano, Italy
Milko T. Marinov	University of Ruse, Bulgaria
Mia Markey	The University of Texas at Austin, USA
Maja Matijasevic	University of Zagreb, Croatia
Rashid Mehmood	School of Engineering, Swansea, UK
Stefania Montani	Università del Piemonte Orientale, Italy
Ramón Moreno Jimenez	Universidad del País Vasco, Spain
Naoto Mukai	Tokyo University of Science, Japan
Christine Mumford	Cardiff University, UK
Jun Munemori	Wakayama University, Japan
Hirofumi Nagashino	The University of Tokushima, Japan
Kazumi Nakamatsu	University of Hyogo, Japan
Zorica Nedic	University of South Australia, Australia
Ngoc Thanh Nguyen	Wroclaw University of Technology, Poland
Vesa A. Niskanen	University of Helsinki, Finland
Lidia Ogiela	AGH & University of Science and Technology, Poland
Yukio Ohsawa	The University of Tokyo, Japan
Gabriel Oltean	Technical University of Cluj-Napoca, Romania
Vasile Palade	Oxford University, UK
Gabriella Pasi	Università degli Studi di Milano Bicocca, Italy
Kunal Patel	Ingenuity Systems, USA
Jose Pazos-Arias	University of Vigo, Spain
Carlos Pedrinaci	The Open University, UK
Alfredo Petrosino	Università di Napoli Parthenope, Italy
Dilip Pratihar	Indian Institute of Technology, India
Goran D. Putnik	University of Minho, Portugal
Minvydas Ragulskis	Kaunas University of Technology, Lithuania
Elisabeth Rakus-Andersson	Blekinge Institute of Technology, Sweden
Nancy Reed	University of Hawaii, USA
Paolo Remagnino	Kingston University, UK
Marina Resta	DIEM sezione di Matematica Finanziaria, Italy
Oleg Rostanin	German Research Center for Artificial Intelligence (DFKI), Germany
Asit Saha	Central State University, USA
Ziad Salem	Aleppo University, Syria
Cesar Sanin	University of Newcastle, Australia
Carlo Sansone	Università di Napoli Federico II, Italy

Mika Sato-Ilic	University of Tsukuba, Japan
Simon Scerri	University of Ireland Galway, Ireland
Benedikt Schmidt	SAP Research, Germany
Klaus Schneider	University of Kaiserslautern, Germany
Steven Schockaert	Ghent University, Belgium
Friedhelm Schwenker	University of Ulm, Germany
Udo Seiffert	Fraunhofer Institute IFF Magdeburg, Germany
Dharmendra Sharma	University of Canberra, Australia
Hisao Shizuka	Kogakuin University, Japan
Christos Sioutis	DSTO, Australia
Masato Soga	Wakayama University, Japan
Margarita Sordo	Harvard University, USA
Anthony Soroka	Cardiff University, UK
Myra Spiliopoulou	Otto-von-Guericke-Universität, Germany
Dipti Srinivasan	National University of Singapore, Singapore
Jadranka Sunde	DSTO, Australia
Nobuo Suzuki	KDDI Corporation , Japan
Edward Szczerbicki	The University of Newcastle, Australia
Kazuhiro Takeda	Shizuoka University, Japan
Hirokazu Taki	Wakayama University, Japan
Tatiana Tambouratzis	University of Piraeus, Greece
Pavel Tichy	Rockwell Automation Research Centre, Czech Republic
Peter Tino	The University of Birmingham, UK
Carlos Toro	VICOMTech, Spain
Maria Torsello	University of Bari, Italy
Edmondo Trentin	University of Siena, Italy
George A. Tsirhrintzis	University of Piraeus, Greece
Kazuhiko Tsuda	University of Tsukuba, Japan
Jeffrey Tweedale	University of South Australia, Australia
Seiichi Uchida	Kyushu University, Japan
Eiji Uchino	Yamaguchi University, Japan
Taketoshi Ushijima	Kyushu University, Japan
Sunil Vadera	University of Salford, UK
Annamarie Varkonyi Koczy	Obuda University, Hungary
István Vassányi	University of Pannonia, Hungary
Alfredo Vellido	Universitat Politècnica de Catalunya, Spain
Juan D. Velásquez	University of Chile, Chile
Maria Virvou	University of Piraeus, Greece
Sara Volpi	University of Pisa, Italy
Junzo Watada	Waseda University, Japan
Toyohide Watanabe	Nagoya University, Japan
Rosina Weber	The iSchool at Drexel, USA
Norbert Wehn	University of Kaiserslautern, Germany
Richard J. White	Cardiff University, UK

M. Howard Williams	Heriot-Watt University, UK
Katsutoshi Yada	Kansai University, Japan
Kunihiro Yamada	Tokai University, Japan
Zijiang Yang	York University, Canada
Hiroyuki Yoshida	Harvard Medical School, USA
Jane You	The Hong Kong Polytechnic University, Hong Kong
Takaya Yuizono	JAIST, Japan
Cecilia Zanni-Merk	LGeCo - INSA de Strasbourg, France

Sponsoring Institutions

Center for Computational and Mathematical Modeling (CM)², University of Kaiserslautern, Germany

German Research Center for Artificial Intelligence (DFKI) GmbH, Kaiserslautern, Germany

Institute of Integrated Sensor Systems, University of Kaiserslautern, Germany

Table of Contents – Part I

Artificial Neural Networks, Connectionists Systems and Evolutionary Computation

Adaptive Recurrent Neuro-Fuzzy Networks Based on Takagi-Sugeno Inference for Nonlinear Identification in Mechatronic Systems	1
<i>Florin Ionescu, Dragos Arotaritei, and Stefan Arghir</i>	
Evolution of Iterative Formulas Using Cartesian Genetic Programming	11
<i>Milos Minarik and Lukas Sekanina</i>	
A New Clustering Algorithm with the Convergence Proof	21
<i>Hamid Parvin, Behrouz Minaei-Bidgoli, and Hosein Alizadeh</i>	
On the Comparison of Parallel Island-Based Models for the Multiobjectivised Antenna Positioning Problem	32
<i>Eduardo Segredo, Carlos Segura, and Coromoto León</i>	
Adaptive Request Distribution in Cluster-Based Web System	42
<i>Krzysztof Zatwarnicki</i>	
Globally Evolved Dynamic Bee Colony Optimization	52
<i>Anggi Putri Pertiwi and Suyanto</i>	
Polytope Classifier: A Symbolic Knowledge Extraction from Piecewise-Linear Support Vector Machine	62
<i>Vilen Jumutc and Andrey Bondarenko</i>	
A Library of Nonlinearities for Modeling and Simulation of Hybrid Systems	72
<i>Florin Ionescu, Dragos Arotaritei, Stefan Arghir, George Constantin, Dan Stefanoiu, and Florin Stratulat</i>	
Cluster Validity Measures Based on the Minimum Description Length Principle	82
<i>Olga Georgieva, Katharina Tschumitschew, and Frank Klawonn</i>	
A Trajectory Tracking FLC Tuned with PSO for Triga Mark-II Nuclear Research Reactor	90
<i>Gürçan Lokman, A. Fevzi Baba, and Vedat Topuz</i>	

Machine Learning and Classical AI

Things to Know about a (dis)similarity Measure	100
<i>Lluís Belanche and Jorge Orozco</i>	
A New Classifier Ensembles Framework	110
<i>Hamid Parvin, Behrouz Minaei-Bidgoli, and Akram Beigi</i>	
Policy Gradient Reinforcement Learning with Environmental Dynamics and Action-Values in Policies	120
<i>Seiji Ishihara and Harukazu Igarashi</i>	
Fuzzy c -Means Clustering with Mutual Relation Constraints: Construction of Two Types of Algorithms	131
<i>Yasunori Endo and Yukihiro Hamasuna</i>	

Adaptive HTTP Request Distribution in Time-Varying Environment of Globally Distributed Cluster-Based Web System	141
<i>Anna Zatwarnicka and Krzysztof Zatwarnicki</i>	

Distributed BitTable Multi-Agent Association Rules Mining Algorithm	151
<i>Walid Adly Atteya, Keshav Dahal, and M. Alamgir Hossain</i>	

Sentiment Analysis of Customer Reviews: Balanced versus Unbalanced Datasets	161
<i>Nicola Burns, Yixin Bi, Hui Wang, and Terry Anderson</i>	

Agent, Multi-Agent Systems, Intelligent Robotics and Control

Agents' Logics with Common Knowledge and Uncertainty: Unification Problem, Algorithm for Construction Solutions	171
<i>Vladimir V. Rybakov</i>	

Labelled Transition System Generation from Alvis Language	180
<i>Leszek Kotulski, Marcin Szpyrka, and Adam Sedziwy</i>	

Snapshot Reachability Graphs for Alvis Models	190
<i>Leszek Kotulski, Marcin Szpyrka, and Adam Sedziwy</i>	

Merging Belief Bases by Negotiation	200
<i>Trong Hieu Tran, Quoc Bao Vo, and Ryszard Kowalczyk</i>	

Semantic Distance Measure between Ontology Concept's Attributes	210
<i>Marcin Pietranik and Ngoc Thanh Nguyen</i>	

Casual Schedule Management and Shared System Using an Avatar	220
<i>Takashi Yoshino and Takayuki Yamano</i>	

Buyer Coalitions on JADE Platform	230
<i>Laor Boongasame, Sakon Wathanathamsiri, and Akawuth Pichanachon</i>	
Emotional Agents in a Social Strategic Game	239
<i>Weiqin Chen, Christoph Carlson, and Mathias Hellevang</i>	
Modelling Multimodal 3D Virtual Environments with Asynchronous Multi-Agent Abstract State Machine	249
<i>Fabio De Felice, Alessandro Bianchi, and Fabio Abbattista</i>	
A Serialization Algorithm for Mobile Robots Using Mobile Agents with Distributed Ant Colony Clustering	260
<i>Munehiro Shintani, Shawn Lee, Munehiro Takimoto, and Yasushi Kambayashi</i>	
A Muscular Activation Controlled Rehabilitation Robot System	271
<i>Erhan Akdoğan and Zeynep Sışman</i>	
Design of a Control System for Robot Shopping Carts	280
<i>Takafumi Kohtsuka, Taishi Onozato, Hitoshi Tamura, Shigetomo Katayama, and Yasushi Kambayashi</i>	
Implementation of the Integrated Management System for Electric Vehicle Charging Stations	289
<i>Seongjoon Lee, Hongkwan Son, Taehyun Ha, Hyungoo Lee, Daekyeong Kim, and Junghyo Bae</i>	

Knowledge Based and Expert Systems

Input-Output Conditions for Automatic Program Generation Using Petri Nets	296
<i>Masahiro Osogami, Teruya Yamanishi, and Katsuji Uosaki</i>	
Representation of Knowledge and Uncertainty in Temporal Logic LTL with Since on Frames Z of Integer Numbers	306
<i>Vladimir V. Rybakov</i>	
KAMET II: KAMET Plus Knowledge Generation	316
<i>Osvaldo Cairó and Silvia Guardati</i>	
Use of Metadata for Access Control and Version Management in RDF Database	326
<i>Kazuhiro Kuwabara and Shotaro Yasunaga</i>	
Knowledge Elicitation Methods Taxonomy: Russian View	337
<i>Tatiana A. Gavrilova, Irina A. Leshcheva, and Maria N. Rumyantseva</i>	

A Formal Framework for Declarative Scene Description Transformation into Geometric Constraints	347
<i>Georgios Bardis, Dimitrios Makris, Vassilios Golfinopoulos, Georgios Miaoulis, and Dimitri Plemenos</i>	
User Movement Prediction Based on Traffic Topology for Value Added Services	357
<i>Marin Vukovic, Dragan Jevtic, and Ignac Lovrek</i>	
Emotion Judgment Method from a Meaning of an Utterance Sentence	367
<i>Seiji Tsuchiya, Misako Imono, Eriko Yoshimura, and Hirokazu Watabe</i>	
A Computationally Efficient Fuzzy Logic Parameterisation System for Computer Games	377
<i>Leslie Jones, Robert Cox, and Sharifa Alghowinem</i>	
Representation and Reuse of Design Knowledge: An Application for Sales Call Support	387
<i>Julian R. Eichhoff and Wolfgang Maass</i>	
Intelligent Vision, Image Processing and Signal Processing	
Generic and Specific Object Recognition for Semantic Retrieval of Images	397
<i>Martin Klinkigt, Koichi Kise, and Andreas Dengel</i>	
A Lane Detection and Tracking Method for Driver Assistance System	407
<i>Nadra Ben Romdhane, Mohamed Hammami, and Hanene Ben-Abdallah</i>	
Effective TV Advertising Block Division into Single Commercials Method	418
<i>Pawel Biernacki</i>	
Robust Depth Camera Based Eye Localization for Human-Machine Interactions	424
<i>Li Li, Yanhao Xu and Andreas König</i>	
Novel Metrics for Face Recognition Using Local Binary Patterns	436
<i>Len Bui, Dat Tran, Xu Huang, and Girija Chetty</i>	
Unsupervised Scene Classification Based on Context of Features for a Mobile Robot	446
<i>Hirokazu Madokoro, Yuya Utsumi, and Kazuhito Sato</i>	

A Novel Emotion Recognizer from Speech Using Both Prosodic and Linguistic Features	456
<i>Motoyuki Suzuki, Seiji Tsuchiya, and Fuji Ren</i>	
Possibilistic Entropy: A New Method for Nonlinear Dynamical Analysis of Biosignals	466
<i>Tuan D. Pham</i>	
Knowledge Management, Ontologies, and Data Mining	
Necessary Tools Choice in a Particular Situation for Computer Conversation	474
<i>Eriko Yoshimura, Misako Imono, Seiji Tsuchiya, and Hirokazu Watabe</i>	
Recommendation and Diagnosis Services with Structure Analysis of Presentation Documents	484
<i>Shinobu Hasegawa, Akihide Tanida, and Akihiro Kashihara</i>	
Topic-Based Recommendations for Enterprise 2.0 Resource Sharing Platforms	495
<i>Rafael Schirru, Stephan Baumann, Martin Memmel, and Andreas Dengel</i>	
Custom Ontologies for an Automated Image Annotation System	505
<i>Gabriel Mihai, Liana Stanescu, Dumitru Dan Burdescu, and Marius Brezovan</i>	
MetaProPOS: A Meta-Process Patterns Ontology for Software Development Communities	516
<i>Nahla Jlaiel and Mohamed Ben Ahmed</i>	
An Ontology-Based Framework for Collaborative Maintenance Planning	528
<i>Ren Genquan, Zhang Yinwen, Zhang Li, Wang Jianmin, and Lan Ting</i>	
A Framework for a Fuzzy Matching between Multiple Domain Ontologies	538
<i>Konstantin Todorov, Peter Geibel, and Céline Hudelot</i>	
Agent-Based Semantic Composition of Web Services Using Distributed Description Logics	548
<i>Mourad Ouziri and Damien Pellier</i>	
Temporal Reasoning for Supporting Temporal Queries in OWL 2.0	558
<i>Sotiris Batsakis, Kostas Stravoskoufouf, and Euripides G.M. Petrakis</i>	

An Algorithm for Finding Gene Signatures Supervised by Survival Time Data	568
<i>Stefano M. Pagnotta and Michele Ceccarelli</i>	
An Empirical Comparison of Flat and Hierarchical Performance Measures for Multi-label Classification with Hierarchy Extraction	579
<i>Florian Brucker, Fernando Benites, and Elena Sapozhnikova</i>	
Towards a Parallel Approach for Incremental Mining of Functional Dependencies on Multi-core Systems	590
<i>Ghada Gasmī, Yahya Slimani, and Lotfi Lakhal</i>	
Paraconsistent Semantics for Description Logics: A Comparison	599
<i>Norihiro Kamide</i>	
Author Index	609

Adaptive Recurrent Neuro-Fuzzy Networks Based on Takagi-Sugeno Inference for Nonlinear Identification in Mechatronic Systems

Florin Ionescu^{1,2}, Dragos Arotaritei³, and Stefan Arghir⁴

¹ HTWG-Konstanz, D-78462 Konstanz, Germany

² Steinbeis Transfer Institute Dynamic Systems, D-10247 Berlin, Germany

³ University of Medicine and Pharmacy "Gr. T. Popa", RO-700115 Iasi, Romania

⁴ University "Politehnica" of Bucharest RO- 060042 Bucharest, Romania

florin.ionescu@stw.de,

{ionescu,darotari,stefan.arghir}@htwg-konstanz.de

Abstract. In this paper we propose a recurrent neuro-fuzzy network (RFNN) based on Takagi-Sugeno inference with feedback inside the RFNN for nonlinear identification in mechatronic systems. The parameter optimization of the RFNN is achieved using a differential evolutionary algorithm. The experimental results are analyzed using a study cases modeled in Simulink: the linear power amplifier and the actuator.

Keywords: neuro-fuzzy network, Takagi-Sugeno, differential evolutionary algorithm, nonlinear identification.

1 Introduction

Many engineering approaches to model-based control or modeling of nonlinearities require an accurate process model. Most of the mechanic, hydraulic and mechatronic systems contain complex nonlinear relations that are difficult to model with conventional data ([1]-[3]). Moreover, when only experimental data is available and a model must be provided, „guessing” and validating can be very difficult and time consuming. The number of trials can be very high. Thus, intelligent systems, such as neural networks and fuzzy systems, which use approaches from biological and human cognitive perspective, can be a solution. These systems have very powerful capabilities of learning and mapping nonlinear properties. They have been studied and applied successfully in problems that involve modeling and controlling of complex, ill-defined or uncertain systems where usually the conventional methods fail to give satisfactory results.

In the case of dynamic systems, the output depends on the past inputs and outputs. This is also the case of the neural networks and neuro-fuzzy systems. Fuzzy systems are known as universal approximators ([4], [5]). If we want to model a dynamic system, which can be nonlinear, we must feed the dynamic system with past inputs and past outputs. The number of delays must be known in advance in all the cases. Also, very long delays can increase substantially the inputs of the system. This leads to the curse of dimensionality problem. Increasing the number of inputs exponentially increases the complexity of the solution.

MATLAB has an architecture and a learning algorithm implemented only for the case of input-output mapping, without recurrence. Simulink has a block that implements the ANFIS in the same manner, without any recurrence. This solution is suitable for controller implementation but the system identification needs much more.

Moreover, if we want to add delays in a recurrent fashion to the ANFIS controller in Simulink, algebraic loops seem not to be permitted by the implementation.

The solution is to use a recurrent neuro-fuzzy network (RFNN) with a learning algorithm adequate for this dynamic architecture ([6]-[18]). The dynamic process that maps not only the current inputs but also the previous ones and possibly previous outputs needs a modified version of the gradient based algorithm used in ANFIS.

The gradient methods proposed by many authors have the disadvantage that they are, in general, very complex, with complicated equations and time consuming implementations. The learning time can be faster than in non-gradient methods, but the coding effort seems to be an argument to search for other non-derivative methods.

We propose an improved approach for a RFNN and we investigate the viability of three learning algorithms with three different approaches: gradient based method [19], hybrid non-derivative method using local search, and differential genetic algorithm ([20], [21]).

In the last case we also present experimental results for a practical case using a differential genetic algorithm. The results are very good and suggest this is a promising direction for future research.

2 Recurrent Fuzzy Neural Network (RFNN)

We must use the feedback connection inside the RFNN. The output delay usually creates difficulties in the Simulink identification diagram, especially when we have more loops. Various authors studied several possibilities of developing a fuzzy recurrent structure. The most used structure is ANFIS, where different feedback loops are added: to internal neurons on different levels, output to inputs, some coefficients, variables or polynomials located in the loop, and others.

We propose a simple architecture, with one input and one output. The feedback loops are internal as shown in Fig. 1. We denoted $v=\{v_{ij}\}$ the parameters that must be adapted, where i and j indicate the location of the respective parameter in the layer and neuron on this layer. Fig. 2 shows the typical ANFIS (Adaptive Neuro Fuzzy Inference System) paradigm.

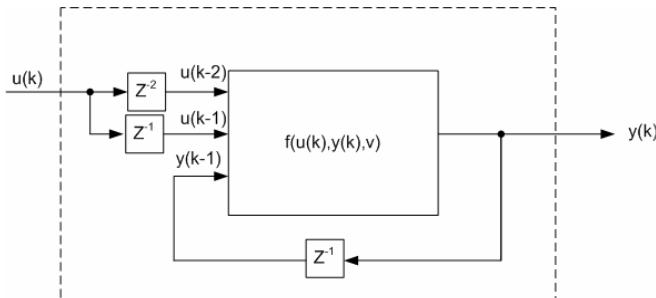


Fig. 1. Recurrent Fuzzy Neural Network (RFNN)

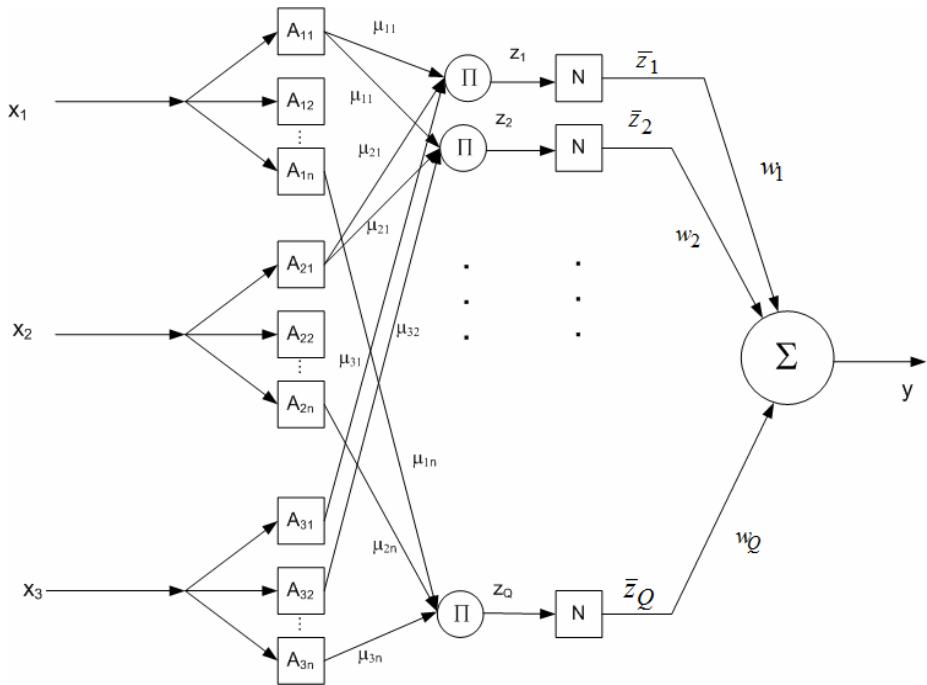


Fig. 2. Detailed RFNN based on ANFIS architecture

A general diagram for RFNNs online (or offline) learning is given in Fig. 3. The input is $u(k)$ and the output is $\hat{y}(k)$. Internally, the RFNN has an ANFIS structure, with delays for inputs and for the feedback $y(k)$.

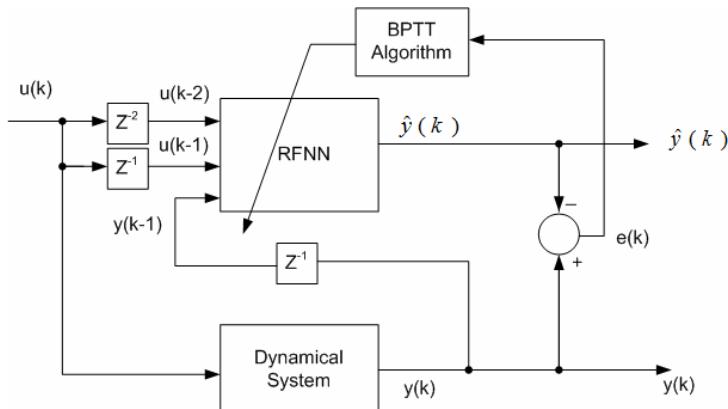


Fig. 3. On-line training diagram for nonlinear identification using RFNN

If we have $m=3$ inputs, the implemented rules have the form:

$$R_q: \text{IF } x_1 \text{ is } A_{1i} \text{ AND } x_2 \text{ is } A_{2j} \text{ AND } x_3 \text{ is } A_{3k} \text{ THEN } y \text{ is } p_1x_1 + p_2x_2 + p_3x_3 + p_4, \quad i,j,k=1\dots n, q=1\dots Q \quad (1)$$

In the form above, n is the number of membership functions (MF) for each one of the $m=3$ inputs. The maximum number of rules is $Q_{\max} = n^m$ and, in this case, $Q_{\max} = n^3$. Every MF is defined as a Gaussian function as in Equation (2). For every input, we initially distributed the Gaussian functions uniformly over the universe of discourse. In order to calculate the output, the ANFIS feed-forward signal propagation is used.

$$\mu_{ij} = e^{-\frac{(x_i - a_{ij})}{2b_{ij}^2}} \quad (2)$$

$$z_q = \mu_{1i_1} \cdot \mu_{2i_2} \cdot \mu_{3i_3} \quad (3)$$

$$\bar{z}_q = \frac{z_q}{\sum_{i=1}^Q z_i} \quad (4)$$

$$y = \sum_{i=1}^Q \bar{z}_i \cdot w_i \quad (5)$$

$$w_i = p_{1i}x_1 + p_{2i}x_2 + p_{3i}x_3 + p_{4i}, \quad i = 1\dots Q \quad (6)$$

In the equations above, as defined with $m=3$ inputs, the index q is given by:

$$q = (i_1 - 1) \cdot n^2 + (i_2 - 1) \cdot n^1 + i_3 \cdot n^0 \quad (7)$$

Prior to the identification procedure, the RFNN must be trained. That means that the RFNN must learn a given trajectory. This trajectory is usually given by experimental data, collected from nonlinear systems to be modelled.

Let us consider a nonlinear process, given by:

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-n_y), u(t-1), u(t-2), \dots, u(t-n_u)) \quad (8)$$

In our case, the model becomes:

$$y(t) = F(y(t-1), u(t-1), u(t-2)) \quad (9)$$

The output error at time t and the error over a trajectory $[t_0, t_1]$ are given by:

$$E(t) = 1/2 \cdot [y(t) - \hat{y}(t)]^2 \quad (10)$$

$$E(t_0, t_1) = \sum_{t=t_0}^{t_1} 1/2 \cdot [y(t) - \hat{y}(t)]^2 \quad (11)$$

The existing studies use the error feedback similar to the NARMAX (Non-linear Autoregressive Moving Average model with exogenous variables) approach. Unlike this approach, we do not use the error feedback.

First we start with a feasible solution by using a teacher-forced version of the ANFIS learning stage. We used a teacher forced variant of the William & Zipser's algorithm for fully recurrent neural network [22]. This is the start solution S for the vector of parameters V . In the second step, we used a new method that combines the trial-and-error and the local search algorithm. $N(S)$ is a set of solutions obtained from S with a slight perturbation of parameters.

$$N = [v_{11}^1, v_{12}^1, \dots, v_{ij}^k, \dots] \quad (12)$$

N represents a vector with n possible directions, where n is the size of N . The perturbation can be applied to one of the n directions of the vector:

$$e_k^p = [0, 0, \dots, 0, 1, 0, \dots, 0] \quad (13)$$

We have two estimations for a sequence of iterations for direction p . Index p indicates the location of the parameter v_{ij}^k :

$$E(v_p + \Delta e_k^p) \quad (14)$$

$$E(v_p - \Delta e_k^p) \quad (15)$$

If one of the solutions (14) or (15) is better than the previous solution, it replaces the current solution and the search continues in this direction. If no improvement is obtained, we go to another direction.

If no direction gives a better response, we decrease the perturbation by a value, usually between 0.1 and 0.5 from the previous one. If no feasible solution is obtained, the algorithm ends, and the current solution is considered optimal.

3 Differential Evolutionary Algorithm for RFNN Optimization

The differential evolutionary (DE) algorithm is a method proposed by Kenneth Price and Rainer Storn in 1994, as a more simple solution than genetic algorithms [18]. DE

is a very simple, population-based, stochastic function optimizer (minimizes the objective). The basic idea behind differential evolutionary (DE) is the scheme for generating trial parameter vectors. The scheme for generating trial vectors uses the vector difference as basic operation for perturbing the population ([18]-[19]).

Each individual from the population is represented by a vector. The vector codes all the parameters of the problem that must be optimized in order to minimize the error. In our case, if we consider a RFNN with $m=3$ inputs, and that each input has $n=3$ Gaussian membership functions, we will have the number of parameter in consequent rules $RT = (m+1) \cdot m^n = 108$ and the total number of vector parameters

$$N_p = 2 \cdot m \cdot n + (m+1) \times m^n = 126 \quad (16)$$

Restrictions apply to the RFNN: inputs in the range $[-1, 1]$ or $[0, 1]$ and the output in the range $[0, 1]$. Also, the central values a_{ij} for Gaussian membership functions belong to the universe of discourse and must be ordered, that is:

$$a_{i1} < a_{i2} < \dots < a_{in}, \quad i = 1, \dots, m \quad (17)$$

All these requirements are inserted as constraints in the DE. Also, each set of parameters is coded as a real value in the range $[U_{low} U_{high}]$, and the vector of parameters has the form presented in Fig. 4.

μ_{11}	μ_{12}	\dots	μ_{ij}	\dots	μ_{nm}	p_{11}	p_{12}	p_{13}	p_{14}	\dots	p_{k1}	p_{k2}	p_{k3}	p_{k4}
------------	------------	---------	------------	---------	------------	----------	----------	----------	----------	---------	----------	----------	----------	----------

Fig. 4. On-line training diagram for nonlinear identification using RFNN

The algorithm is described in short below.

Step 0. Generate the population

Step 1. Chose target vector and the base vector

Step 2. Randomly choose two population members (uniform distribution)

Step 3. Compute the weighted difference vector between target vector and base vector

Step 4. Add the base vector

Step 5. Compute the next vector:

$$x_{1,k+1} = u_{1,k} \quad \text{if } f_o(u_{1,k}) \leq f_o(x_{1,k+1}), \text{ else } x_{1,k+1} = x_{1,k} \quad (18)$$

After step 5, a new population is generated, $P_{x,k+1}$. In Fig. 5 the algorithm is presented synthetically.

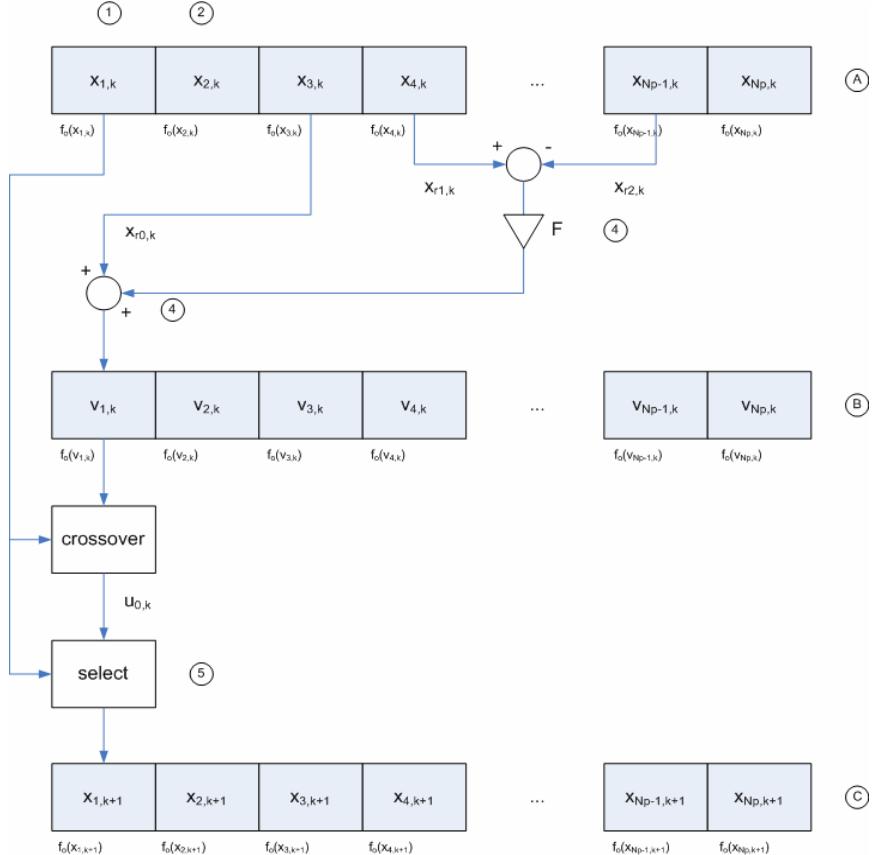


Fig. 5. Differential evolutionary algorithm for one generation, operations between two vectors. A represents the initial population, $P_{x,k}$, at that is population at k generation, B represents the mutant population $P_{m,k}$ and C represents the new population $P_{x,k+1}$.

4 Experimental Results

Fig. 6 depicts a Simulink diagram of a linear force amplifier. In this diagram, the actuator is emphasized by the dotted rectangle. In our approach we want to model the nonlinear system represented by the actuator, using the proposed RFNN structure. The model uses the input-output data set, obtained from the classic model as $(Qv(t), xc(t))$ pairs in discrete space, 1000 points and $t=2$ sec.

The parameters of RFNN are: $n=3$ and $m=3$ inputs. The DE uses a population of 100 vectors and the number of iterations is set to 100.

The sum of all the errors in the learning stage during the trajectory over 1000 points, in the last generation, for the best vector, is $E_{\text{total}} = 1,24 \cdot 10^{-3}$. The error in the test case over 1000 steps is $8.94 \cdot 10^{-3}$, which is an excellent result.

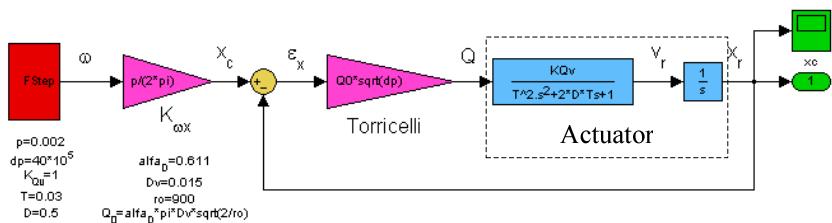


Fig. 6. The linear power amplifier

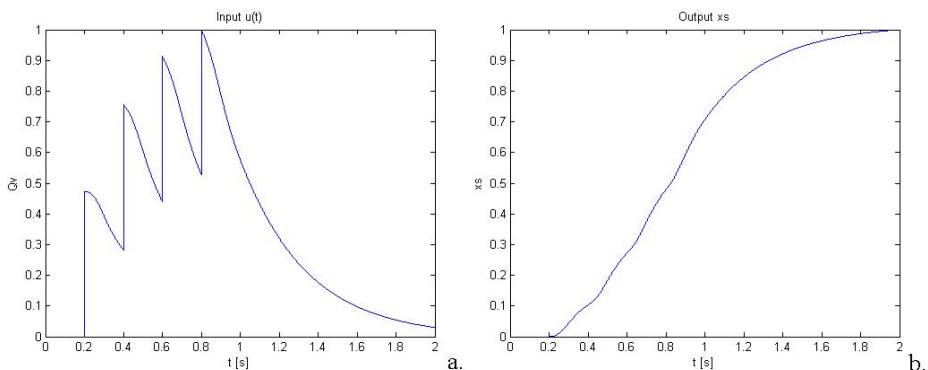


Fig. 7. Input $u(t)$ (a.) and output $x_s(t)$ (b.)

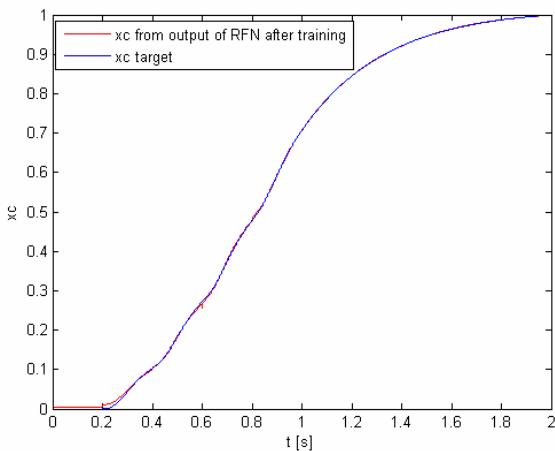


Fig. 8. The target and the output of RFNN in the test case

5 Conclusions

We proposed a differential evolutionary algorithm to optimize an RFNN architecture with feedback in a Simulink diagram. The optimizations are made using a dynamic approach, using the trajectory of input-output signals.

The experimental results use a practical application, the model of an actuator from a linear power amplifier used in mechatronic systems. The results are very good and the approximation of the nonlinear plant by the proposed RFNN is very good.

The results of the proposed approach, using the differential genetic algorithm, are very good and suggest that this is a viable development direction for future research on more complex recurrent architectures.

Acknowledgments. This work was funded by the DFG, Germany and, for S. Arghir was supported by the Romanian Ministry of Labor, Family and Social Protection through the Financial Agreement POSDRU/6/1.5/S/19. Also, the authors would like to thank the University of Sciences HTWG-Konstanz, Germany for its support, to the Department of Mechatronics and to the Institute of Applied Research (IAF).

References

1. Stratulat, F., Ionescu, F.: Linear Control Systems. Steinbeis-Edition (2009)
2. Ionescu, F.: Nonlinear Mathematical Behaviour and Modelling of Hydraulic Drive Systems. In: Proceedings of the 2nd World Congress of Nonlinear Analysts, vol. 30, part 3, pp. 1447–1461. Pergamon Press, Athens (1996)
3. Arotaritei, D., Ionescu, F., Constantin, G., Stratulat, F., Arghir, S.: Nonlinear Mathematical Models for Mechanics, Hydraulics, Pneumatics and Electric Systems - Analytical and Neuro-Fuzzy Approaches. DFG-Research Report. HTWG-Konstanz, Germany (2010)
4. Buckley, J.J.: Sugeno type controllers are universal controllers. *Fuzzy Sets and Systems* 53, 299–303 (1993)
5. Castro, J.L., Delgado, M.: Fuzzy systems with defuzzification are Universal approximators. *IEEE Transactions on Systems, Man and Cybernetic* 26(1), 149–152 (1996)
6. Chak, C.K., Feng, G., Ma, J.: An adaptive fuzzy neural network for MIMO system model approximation in high-dimensional spaces. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics* 28(3), 436–446 (1998)
7. Gonzalez-Olvera, M.A., Tang, Y.: A new recurrent neuro-fuzzy network for identification of dynamic systems. *Fuzzy Sets and Systems* 158, 1023–1035 (2007)
8. Gonzalez-Olvera, M.A., Tang, Y.: Nonlinear System Identification and Control Using an Input-Output Recurrent Neurofuzzy Network. In: Proceedings of the 17th World Congress The International Federation of Automatic Control, Seoul, Korea, pp. 7480–7485 (2008)
9. Gorrini, V., Bersini, H.: Recurrent Fuzzy Systems. In: Proceedings of the 3rd Conference on Fuzzy Systems (FUZZ-IEEE 1994), pp. 193–198 (1994)
10. Graves, D., Pedrycz, W.: Fuzzy prediction architecture using recurrent neural networks. *Neurocomputing* 72, 1668–1678 (2009)
11. Yu, W.: State-Space Recurrent fuzzy neural networks for nonlinear system identification. *Neural Processing Letters* 22, 391–404 (2005)

12. Juang, C.F.: A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms. *IEEE Transactions on Fuzzy Systems* 10(2), 155–170 (2002)
13. Lee, C.H., Teng, C.C.: Identification and control of dynamic systems using recurrent fuzzy neural networks. *IEEE Trans. Fuzzy Syst.* 8(4), 349–366 (2000)
14. Savran, A.: An adaptive recurrent fuzzy system for nonlinear identification. *Applied Soft Computing* 7, 593–600 (2007)
15. Abdelrahim, E.M., Yahagi, T.: A new transformed input-domain ANFIS for highly nonlinear system modeling and prediction. In: Canadian Conference on Electrical and Computer Engineering, vol. 1, pp. 655–660 (2001)
16. Babuška, R., Verbruggen, H.: Neuro-fuzzy methods for nonlinear system identification—review. *Annual Reviews in Control* 27, 73–85 (2003)
17. Baruch, et al.: A fuzzy-neural multi-model for nonlinear systems, identification and control. *Fuzzy Sets and Systems* 159, 2650–2667 (2008)
18. Baruch, I., Gortcheva, E., Thomas, F., Garrido, R.: A Neuro-Fuzzy model for nonlinear plants identification. In: Proceedings of the IASTED International Conference Modeling and Simulation MS 1999, pp. 326–331 (1999)
19. Bersini, H., Gorriani, V.: A simplification of the backpropagation-through-time algorithm for optimal neurocontrol. *IEEE Trans. Neural Networks* 8(2), 437–441 (1997)
20. Price, K., Rainer, S., Jouni, L.: Differential Evolution - A Practical Approach to Global Optimization. Springer, Heidelberg (2005)
21. Chakraborty, U.: Advances in Differential Evolution. SCI. Springer, Heidelberg (2008)
22. Williams, R.J., Zipser, D.: A learning algorithm for continually running fully recurrent neural networks. *Neural Computing* 1(2), 270–280 (1989)

Evolution of Iterative Formulas Using Cartesian Genetic Programming

Milos Minarik and Lukas Sekanina

Brno University of Technology, Faculty of Information Technology,
Božetěchova 2, 612 66 Brno, Czech Republic
`{iminarikm,sekanina}@fit.vutbr.cz`

Abstract. Many functions such as division or square root are implemented in hardware using iterative algorithms. We propose a genetic programming-based method to automatically design simple iterative algorithms from elementary functions. In particular, we demonstrated that Cartesian Genetic Programming can evolve various iterative formulas for tasks such as division or determining the greatest common divisor using a reasonable computational effort.

1 Introduction

Genetic programming (GP) that has been intensively developed since the early 1990's [1, 2], is now considered as a robust method allowing automated problem solving in many scientific and engineering areas [3]. Symbolic regression which enables to obtain a symbolic expression from a given data, is considered as one of main application domains of genetic programming [4]. In the recent years, there have been several attempts to evolve solutions that can iteratively be executed to produce potentially infinite sequences of results (see a survey in [5]). This may be useful in the applications such as generative and developmental systems, self-modifying systems and approximation of functions.

With the inspiration in our previous work on evolutionary design of arbitrarily large sorting networks [6], we will consider a different scenario for iterative formula evolution in this paper. The goal is to investigate to what extent a GP system is capable of evolving the program P which can approximate a given target function when applied iteratively on its own result, i.e.

$$x_1^{i+1} \dots x_n^{i+1} = P(x_1^i \dots x_n^i) \quad (1)$$

where $x_1^i \dots x_n^i$ is the i -th approximation of a solution and n is the problem dimension. Since this kind of computations is often carried out in hardware to compute division, square root or goniometric functions, P will be sought in a form that can easily be implemented in hardware, ideally using components such as adders, subtractors, shifters and multiplexers. This may be viewed as a constraint with respect to the standard analytical methods that can utilize a full repertoire of functions. Another important issue is ensuring a reasonable convergence of P .

In order to evolve iterative algorithms, we will utilize Cartesian Genetic Programming (CGP) which is suitable not only for symbolic regression but also for digital circuit design [7]. CGP will be used in its basic version (as presented in [7]) although advanced versions such as self-modifying CGP [8] have already been developed and applied to function approximation. Since our ultimate goal is to evolve new iterative algorithms for a hardware implementation, the approach utilized in self-modifying CGP is not suitable (see Section 2.2). The proposed approach will be evaluated whether it is able to re-discover three iterative algorithms – Euclidean algorithm, Newton-Raphson division and Goldschmidt division. We will investigate the computational effort needed for solving these tasks.

The rest of the paper is organized as follows. After introducing the concept of Cartesian genetic programming in Section 2, the proposed method will be presented in Section 3. Section 4 summarizes the results obtained using our three case studies. Finally, conclusions are given in Section 5.

2 Cartesian Genetic Programming

2.1 Standard CGP

In the standard CGP [7, 9], a candidate program is represented as an array of n_c (columns) $\times n_r$ (rows) of programmable nodes. The number of inputs, n_i , and outputs, n_o , is fixed. Each node can be connected either to the output of a node placed in previous l columns or to one of program inputs. Setting of the l parameter allows to control the level of interconnectivity. Feedback is not allowed. Each node is programmed to perform one of n_a -input functions defined in the set Γ (n_f denotes $|\Gamma|$). Each node is encoded using $n_a + 1$ integers where values $1 \dots n_a$ are indexes of the input connections and the last value is the function code. Every program is encoded using $n_c \cdot n_r \cdot (n_a + 1) + n_o$ integers. Figure 1 shows the example of a candidate program and its chromosome.

The initial population is constructed either randomly or by a heuristic procedure. Every new population consists of the best individual of the previous population and its λ offspring individuals, i.e. the $(1 + \lambda)$ strategy is employed. The offspring individuals are created using a point mutation operator which modifies h randomly selected genes of the chromosome, where h is a user-defined value. For symbolic regression problems, the goal is usually to minimize the mean absolute error of the candidate program response Y and target response T .

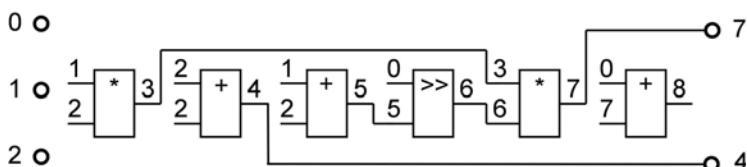


Fig. 1. Example of a program in CGP with parameters: $n_c = 6$, $n_r = 1$, $n_i = 3$, $n_o = 2$, $l = 6$, $n_a = 2$, $\Gamma = \{+ (0), * (1), \text{shift} (2)\}$. Chromosome: 1, 2, **1**, 2, 2, **0**, 1, 2, **0**, 0, 5, **2**, 3, 6, **1**, 0, 7, **0**, 7, 4. (Node function is typed in bold).

2.2 Extensions of CGP

Modular CGP (MCGP) or Embedded CGP (ECGP) [10, 11] utilizes the concept of evolution and acquisition of Automatically Defined Functions (ADFs). Modules can then be replicated as a whole block offering a way to find solutions to some problems of modular nature, e.g. even parity. The main disadvantage of this approach (in terms of the evolutionary design of iterative formulas for hardware) is a rapid growth of a phenotype, because the modules cannot be used in an iterative manner.

Self-modifying CGP (SMCGP) [5, 8] enables the use of special kind of nodes – so called self-modification nodes. These nodes do not affect the genotype, but they are utilized during the phenotype execution. Although this CGP extension can generate virtually arbitrarily large solutions, these solutions are not feasible for hardware implementation, because of the phenotype growth during the iterations.

3 Proposed Method

The proposed method is based on a standard CGP, however several modifications have been introduced.

3.1 Iterative Application of Phenotype

The aim of the evolution is to find a program that approximates the target n -dimensional function, which computes the results of the $i + 1$ th iteration using the results of the i th iteration. Repeating this process for a specified number of iterations we get the final output values. Given these preconditions some assumptions considering the number of inputs (n_i) and number of outputs (n_o) of the chromosome can be made. If n_o is greater than n_i then some of the outputs will not be used. In the other case, when n_i is greater than n_o , some of the inputs will be unnecessary. Considering given facts we will assume that $n_i = n_o$.

3.2 Modifications of CGP

The representation of a chromosome is the same as in the standard CGP. A 32-bit signed value is used to represent numbers. Node functions ($V^{n_a} \rightarrow V$) should be chosen so they are easily implementable in hardware. This implies the use of basic operations only (addition, subtraction, multiplication, bit shift and multiplexing). To represent the node returning a constant value the representation of the node has to be changed in such way that it may contain a parameter with a constant value used as an output disregarding the inputs.

As it has been mentioned in Section 2 the fitness function is typically defined as the mean absolute error of the CGP output and target output. However, considering the proposed modification, the fitness function can utilize the outputs of individual iterations. These outputs should not be directly compared with correct outputs of particular iterations, because this would give an explicit information

about the searched solution. However we can use the evaluated outputs of the iterations to check for convergence, because many iterative algorithms get closer to the correct solution by each iteration. The fitness function can therefore be defined, for example, to minimize a sum of ratios of differences between program response Y and target response T over the specified number of iterations n_{it}

$$\sum_{i=1}^s \sum_{j=1}^{n_{it}} \frac{|Y_{ij} - T_{ij}|}{|Y_{ij-1} - T_{ij-1}|} \quad (2)$$

where s is the number of test points (input/output pairs). Note that primary test inputs are taken as Y_{i0} when computing the ratio after the first iteration. The main advantage of such fitness function is the fact, that it expresses the speed of convergence. It can be seen that equation 2 determines how many times the approximation of one iteration was better than the subsequent iteration.

Genetic operators are used with the probability P_m (for mutation) and P_c (for crossover). The operators are applied using the following procedure:

1. Given the previous evaluated generation, select two parents using tournament selection.
2. Apply the crossover to the parents with probability P_c .
3. Apply the mutation to created offspring with probability P_m .
4. Insert the offspring into the new generation and go back to 1.

Mutation can change the chromosome at the level of an individual node. The affected node is randomly selected. In addition, the mutation can change node's function, connection or internal parameter. Crossover is implemented as one point crossover and applied at the level of nodes.

3.3 Test Problems

The proposed method was tested on three problems. First of the algorithms chosen was the Euclidean algorithm for finding the greatest common divisor (GCD) of two numbers (a, b). It is based on the principle that the GCD does not change, if smaller of the numbers is subtracted from the larger. The algorithm can be written as a sequence of equations

$$a = q_0 b + r_0$$

$$b = q_1 r_0 + r_1$$

$$r_0 = q_2 r_1 + r_2$$

$$r_1 = q_3 r_2 + r_3$$

...

where a and b stand for the original values for which the GCD has to be calculated. Symbols q and r represent quotients and remainders respectively.

The training set in this case should contain several pairs of relatively prime positive integers to avoid the situation, where one of the numbers is the divisor of the other number and therefore it is directly the GCD. In such case one of the inputs would be the correct output and even the solutions, which are just passing the inputs to outputs without any modifications could achieve higher fitness (which is not desirable).

Next two test problems are iterative division algorithms which are supposed to find a quotient Q of N and D ($Q = N/D$).

In the first iterative division algorithm (Newton-Raphson method), there is actually no division involved. This algorithm first finds the reciprocal of the divisor D and then multiplies it by N to find the resulting quotient. The reciprocal can be found iteratively using the expression $X_{i+1} = X_i(2 - DX_i)$, where $\frac{1}{2} \leq D \leq 1$. The aim of our experiment is therefore finding a reciprocal of the divisor lying in the specified range by means of iterative algorithms. Training set for this case will contain randomly chosen numbers from the forementioned range. The numbers are chosen at the beginning of each run and do not change their values during evaluation.

The last problem is the Goldschmidt division which iteratively multiplies dividend and divisor by the same value, i.e.

$$Q = \frac{N}{D} \frac{F_1}{F_2} \frac{F_2}{F_3} \cdots \frac{F_n}{F_n}.$$

When the divisor converges to 1, the dividend is equal to the quotient. There are also some conditions that have to be fulfilled. First of all the input needs to be modified in such way, that $0 < D < 1$. Using the binary encoding of the values this is a simple task, because it is sufficient just to shift the input values. When these conditions are satisfied, the factor F_i can be computed as $F_{i+1} = 2 - D_i$. The training set in this case is the set of pairs of numbers (N, D) lying in the specified range. Training set is randomly chosen at the beginning of each run and is not changed during evaluation.

3.4 Computational Effort

In order to assess the effectiveness of our search algorithm and setting of control parameters, the computational effort will be used as a measure [2].

The computational effort is the number of individual chromosomes evaluations needed to find a solution to give a certain probability of success. As a result of an experiment there will be (N_s) successful runs out of the total number of runs (N_t) . Given the generation numbers, on which the solutions were found in individual runs we can compute instantaneous probability $Y(M, i)$, where M is the population size and i is the number of generation. The value of $Y(M, i)$ represents the probability of finding a solution in given generation. Using these values we can compute the cumulative probability of success $P(M, i)$. Afterwards the number of independent runs $R(M, i, z)$ required for achieving the probability z of finding the solution by generation i can be computed. Given the values of

$R(M, i, z)$ the number of individuals that must be processed $I(M, i, z)$ in order to yield a solution with probability z can be computed as $I(M, i, z) = M(i+1)R(z)$. Notice the use of $i + 1$ as the number of generation. It is used to take into account the initial population. The computational effort E is the minimal value of $I(M, i, z)$ over all the generations. The first generation where $I(M, i, z)$ reaches the global minimum is denoted as i^* and the computational effort can thus be computed as

$$E = I(M, i^*, z) = M(i^* + 1)R(z). \quad (3)$$

4 Results

We performed a number of experiments to determine the computational effort and a suitable setting of CGP parameters for our test problems. If not stated otherwise, the following set of node functions is applied: identity (ID), plus (PLUS), minus (MINUS), multiplication (MULT), constant (CONST) and multiplexer (MUX). In order to evaluate a particular setting of CGP parameters, 200 independent runs were conducted.

4.1 Newton-Raphson Iterative Division

It was recognized that the solution must contain at least two inputs, therefore we can put $n_i = n_o = 2$. One input is used for passing the intermediate result, while the other one holds the original divisor. The probabilities of crossover and mutation were set to $P_c = 0.5$ and $P_m = 0.5$ respectively. The number of individuals was chosen as $M = 1000$, the maximal number of generations $G = 10000$, $n_{it} = 5$ and $s = 8$. Table 1 gives the computational effort for various setting of n_c and n_r . According to the results, the best combination is $n_c = 5$, $n_r = 3$.

Table 1. Computational effort regarding the number of columns and rows

n_c, n_r	Computational effort
3, 2	$2.24 \cdot 10^8$
5, 3	$1.74 \cdot 10^7$
7, 4	$3.64 \cdot 10^7$
9, 4	$2.72 \cdot 10^7$

Next step was choosing the probabilities of mutation and crossover. When finding each of the probabilities the other probability was set to 0, so it could not affect the results. The results are summarized in Figure 2. We used $n_c = 5$, $n_r = 3$.

As can be seen the mutation probability significantly influences the computational effort and the best value is $P_m = 0.6$. The crossover does not seem to affect the computational effort very much. It can be noticed that values of the computational effort are significantly higher for the crossover probability from 0.1 to 0.3. This is probably caused by the fact that the mutation probability

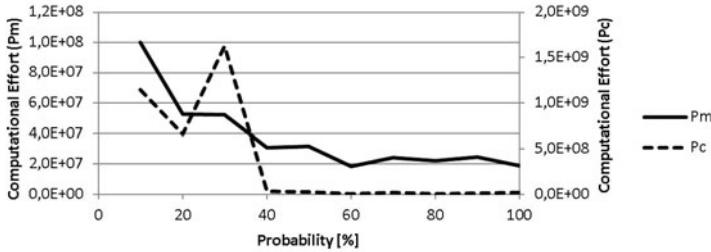


Fig. 2. Computational effort vs mutation probability and crossover probability

was set to 0 in this experiment. Therefore, there was a big portion of individuals in each generation that was not modified, so the evolution was slow compared to higher values of crossover probability. On the other hand after reaching 0.4 the crossover probability does not seem to influence the computational effort anymore. Considering this the crossover probability was chosen as $P_c = 0.05$ because of the little affect it has on evolution.

Finally, we experimented with the population size. The CGP parameters were set to values already found and the only variable parameters were the population size M and maximum number of generations G . The maximum number of generations was chosen in such way that the maximum number of individual evaluations was constant for different values of M . Table 2 shows that $M = 100$ is the most suitable setting.

Table 2. Computational effort vs population size

Population size	Computational Effort
10	16,058,949
20	5,613,264
50	5,805,492
100	5,038,127
200	5,727,632
500	9,250,326
1000	24,363,608

The fitness function uses the results of all the iterations to check for convergence as stated in Section 3. Finally, the stop condition limiting the number of iterations must be chosen. Considering the specification of the experiment there are two major ways of defining the stop condition. One of them is obviously defining the maximum number of iterations. The other one is to check the difference between outputs of subsequent iterations and if the difference is smaller than some predefined value, the evaluation is stopped. The second condition will work, if the solution is correct, so the results of subsequent iterations converge. But if the solution is not correct, the results can be constant or can even diverge. These situations could be omitted by checking the convergence. In this experiment, the fixed number of iterations was used.

Having chosen the CGP parameters we made 200 runs to find out that CGP terminates with a correct solution in 52 % of runs and with the computational effort 5,038,127. One of the solutions found is depicted in Figure 3.

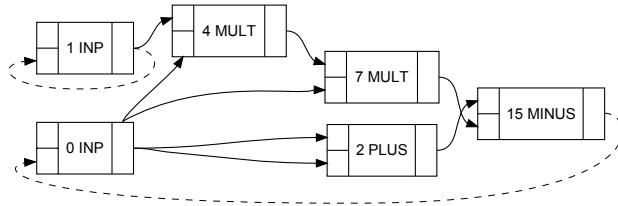


Fig. 3. Evolved iterative division expression. The connections between outputs and inputs are shown as dashed arcs.

The generality of the solution can be determined by comparing its structure to the original algorithm. The solution shown is obviously representing the original algorithm in expanded form $X_{i+1} = 2X_i - DX_i^2$. Other solutions found had almost the same structure.

4.2 Goldschmidt Division

In this experiment, we could not determine any CGP settings leading to the solution. After an investigation we found out, that the evolution had problems finding the constant 2. This problem arises due to the nature of searched expression. The problem is that even if the constant differs from 2 just slightly, the result will differ a lot. Therefore we have limited the set of constants just to natural numbers. Using this restrictions, several solutions were discovered. The best-performing setting is: $n_r = 2$, $n_c = 5$, $n_i = n_o = 2$, $M = 500$, $G = 10000$, $P_m = 0.8$, $P_c = 0.05$, $n_{it} = 10$ and $s = 8$.

In summary, 15 out of 100 runs were successful and the computational effort was 4,811,211. One of the solutions found is depicted in Figure 4. It is identical with the original algorithm and optimal.

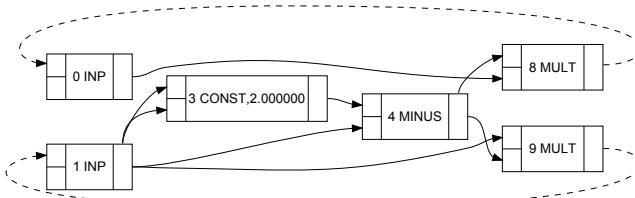


Fig. 4. Optimal solution for Goldschmidt iterative division. The connections between outputs and inputs are shown as dashed arcs.

4.3 Euclidean Algorithm

The most suitable CGP parameters were determined the same way as in the previous experiments. Their values are: $n_r = 3$, $n_c = 5$, $n_i = n_o = 2$, $M = 500$, $G = 2000$, $P_m = 0.6$, $P_c = 0.05$, $n_{it} = 40$ and $s = 20$. Although the proposed method enables the use of results from individual iterations, it has not proved useful in this case. Therefore, the fitness function used just the overall results out of 20-point training set. The 91 % of runs led to a perfectly working solution (with the computational effort of 2,086,497).

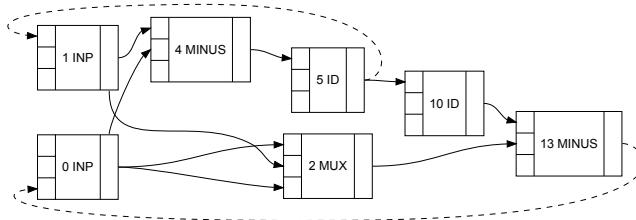


Fig. 5. Example of an individual found for Euclidean algorithm. The connections between outputs and inputs are shown as dashed arcs.

5 Conclusions

We have shown that the standard CGP is capable of evolving simple iterative formulas. Based on training data, general well-known iterative formulas were automatically rediscovered for tasks of division and GCD. In comparison with classical symbolic regression problems [2], the computational effort increased by approximately one order of magnitude. This finding seems to be justifiable since determining a suitable iterative formula is a more difficult task than the classic symbolic regression. Direct comparison with other methods of evolutionary iterative algorithms design is quite difficult due to the fact the other authors used different sets of experiments. Although we have not forced CGP to minimize the size of phenotype we obtained almost optimal solutions with respect to the constraints given.

In our future work, we plan to evolve iterative formulas for other functions (square root, exponential etc.) and simultaneously to consider various trade-offs that are typical for hardware, for example, a limited set of functions (no multiplication), limited interconnection possibilities (to support pipelining) or reusing of components vs convergence and precision.

Acknowledgments. This work was partially supported by the grant Natural Computing on Unconventional Platforms GP103/10/1517, the FIT grant FIT-11-S-1 and the research plan Security-Oriented Research in Information Technology, MSM0021630528.

References

- [1] Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge (1992)
- [2] Koza, J.R.: Genetic Programming II: Automatic Discovery of Reusable Programs. MIT Press, Cambridge (1994)
- [3] Koza, J.R.: Human-competitive results produced by genetic programming. *Genetic Programming and Evolvable Machines* 11, 251–284 (2010)
- [4] Schmidt, M.D., Lipson, H.: Coevolution of Fitness Predictors. *IEEE Transactions on Evolutionary Computation* 12, 736–749 (2008)
- [5] Harding, S., Miller, J.F., Banzhaf, W.: Developments in cartesian genetic programming: self-modifying cgp. *Genetic Programming and Evolvable Machines* 11, 397–439 (2010)
- [6] Sekanina, L., Bidlo, M.: Evolutionary design of arbitrarily large sorting networks using development. *Genetic Programming and Evolvable Machines* 6, 319–347 (2005)
- [7] Miller, J.F., Thomson, P.: Cartesian Genetic Programming. In: Poli, R., Banzhaf, W., Langdon, W.B., Miller, J., Nordin, P., Fogarty, T.C. (eds.) *EuroGP 2000*. LNCS, vol. 1802, pp. 121–132. Springer, Heidelberg (2000)
- [8] Harding, S., Miller, J.F., Banzhaf, W.: Self modifying cartesian genetic programming: Parity. In: *2009 IEEE Congress on Evolutionary Computation*, pp. 285–292. IEEE Press, Los Alamitos (2009)
- [9] Miller, J.F., Job, D., Vassilev, V.K.: Principles in the Evolutionary Design of Digital Circuits – Part I. *Genetic Programming and Evolvable Machines* 1, 8–35 (2000)
- [10] Walker, J.A., Miller, J.F.: The Automatic Acquisition, Evolution and Re-use of Modules in Cartesian Genetic Programming. *IEEE Transactions on Evolutionary Computation* 12, 397–417 (2008)
- [11] Kaufmann, P., Platzner, M.: Advanced techniques for the creation and propagation of modules in cartesian genetic programming. In: Proc. of Genetic and Evolutionary Computation Conference, GECCO 2008, pp. 1219–1226. ACM, New York (2008)

A New Clustering Algorithm with the Convergence Proof

Hamid Parvin, Behrouz Minaei-Bidgoli, and Hosein Alizadeh

School of Computer Engineering,
Iran University of Science and Technology (IUST),
Tehran, Iran
`{parvin, b_minaei, halizadeh}@iust.ac.ir`

Abstract. Conventional clustering algorithms employ a set of features; each feature participates in the clustering procedure equivalently. Recently this problem is dealt with by Locally Adaptive Clustering, LAC. However, like its traditional competitors the LAC method suffers from inefficiency in data with unbalanced clusters. In this paper a novel method is proposed which deals with the problem while it preserves LAC privilege. While LAC forces the sum of weights of the clusters to be equal, our method let them be unequal. This makes our method more flexible to conquer over falling at the local optimums. It also let the cluster centers to be more efficiently located in fitter places than its rivals.

Keywords: Subspace clustering, Weighted Clusters, Features Weighting.

1 Introduction

Data clustering or unsupervised learning is an essential and also an ill-posed, NP-hard problem. The objective of clustering is to group a set of unlabeled objects into homogeneous groups or clusters which are overall called a data partition or partitioning [6-7] and [9]. Clustering methods are applied to a set of patterns to partition the set into some clusters such that all patterns within a cluster are similar to each other and different from the members of other clusters. In other words, it is intended to have minimum inter-cluster variances and maximum between-cluster variances [14].

The partitional clustering algorithms such as k-means and k-medoid use exactly one point as the representative of each cluster. They partition the set of input patterns into non-overlapping clusters. They consider one point per cluster and iteratively update the representative of the cluster by placing it at the average (medoid) of the cluster. There is a proof for their convergence to a local minimum. The most well-known partitional clustering algorithm, k-means, is discussed by Jain and Dubes in detailed theoretically [9]. The term "k-means" was first used by James MacQueen [12].

The curse of dimensionality is an ever challenge in all supervised and also unsupervised learning methods. In high dimensional spaces, it is highly likely that, for any given pair of points within the same cluster, there exist at least a few dimensions on which the points are far apart from each other. As a consequence, distance functions that equally use all input features may not be effective. Furthermore, it is very likely that each cluster in a real dataset is correlated with a subset of features meanwhile some others are correlated with other subsets of features. There are three ways to deal

with the curse of dimensionality. One: it could be addressed by requiring the user to specify a subspace [16]. Two: Another way is to reduce the dimensionality of the input space by feature reduction algorithms like PCA. Three: and finally, it can be dealt with by LAC (Locally Adaptive Clustering) algorithm. The first is error-prone, while the second is not fit in some cases and also not feasible in some other cases [4-5]. The LAC algorithm firstly developed by Domeniconi, has been shown that is suitable to solve the problem to a great extent.

Although the third method conquers the problem of imbalanced feature-variance of each cluster, it is not capable of solving the imbalanced inter-cluster variances. Two other methods also are not capable of solving it. It is very common to face a dataset including some dense clusters as well as some sparse ones. However, all the previous methods are not capable of managing the problem. While if the clustering mechanism can take into account the densities of clusters simultaneously with the importance of each feature for them, then the mechanism can overcome the problem. In this paper a novel clustering algorithm is proposed which assigns an importance weight to each cluster as well as a weight vector to all features per each cluster. Each feature along which a cluster is loosely correlated receives a small weight; consequently, the feature becomes less participant in distance function for that cluster than the others. In contrary, each feature along which data are strongly correlated receives a large weight which results in participating in distance function with more effect. Our method benefits from LAC and also uses the cluster importance concurrently.

Our contributions are four-fold.

1. We propose a novel clustering error criterion which takes into consideration the inter-cluster variances.
2. We propose a novel clustering algorithm using the proposed criterion.
3. We prove the convergence of the algorithm.
4. We support our method with some results over a number of real datasets.

2 Related Work

Subspace clustering algorithms are considered as extensions of feature selection methods. They attempt to find clusters in different subspaces of the same dataset. Like feature selection methods, subspace clustering algorithms require a search method and an evaluation criteria. Indeed, the subspace identification problems are related to the problem of finding quantitative association rules that also identify interesting regions of various attributes [13] and [17]. If we consider only subsets of the features, i.e., subspaces of the data, then the clusters that we find can be quite different from one subspace, i.e., one set of features, to another. In addition, the subspace clustering must somehow limit the scope of the evaluation criteria so as to consider different subspaces for each different cluster [16].

Subspace clustering algorithms are divided into two categories according to their searching approaches for subspaces. A brute force approach might be to search through all possible subspaces and apply cluster validation techniques to determine the subspaces with the best clusters [9]. This is not feasible because the subset generation problem is intractable [1] and [10]. Another choice of search technique is heuristic based approaches. This categorization is demonstrated hierarchically in Fig 1 [16].

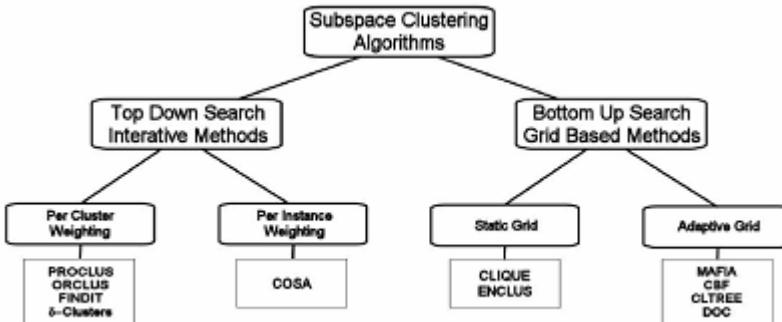


Fig. 1. Categorization subspace clustering algorithms

As shown in Fig 1, the first division in the hierarchy splits subspace clustering algorithms into two groups, the top-down search methods [8] and [11] and bottom-up search methods. The second division is based on the way that the algorithms apply a measure of locality with which they are to evaluate subspaces.

3 Proposed Criterion of Clustering

Consider a set of points in some space of dimensionality D . A weighted cluster C_i is a subset of data points, together with a vector of weights $w_i = (w_{i1}, \dots, w_{iD})$, and also a co-efficient d_i such that the points in C are closely clustered according to the L_2 norm distance weighted using w_i and applied co-efficient d_i . The component w_{ij} measures the degree of participation of feature j to the cluster C_i . Where d_i is diameter of cluster i , in other words, weight of cluster C_i . If the points in C_i are well clustered along feature j , w_{ij} is large, otherwise it is small. Also for d_i , if the cluster C_i is a big cluster, d_i will be small number. It means if a cluster is big (or has high variance), its distances will be degraded, and otherwise they will be enlarged. Clustering algorithm now faces with the problem "how to estimate the weight vector w for each cluster and co-efficients vector d for clusters of dataset".

Now, the concept of cluster is not based only on points, but also involves a weighted distance metric, i.e., clusters are discovered in spaces transformed by w and simultaneously finding the volumes of clusters. For each cluster we now have a w vector reflecting the importances of features in the cluster and also a weight d standing as its diameter. The effect of w is to transform distances so that the associated cluster is reshaped into a dense hypersphere of points separated from other data. While the weight d is to transform data so that the clusters with low diversities become denser and clusters with high diversities become bulkier. In traditional clustering, the partition of a set of points is induced by a set of representative vectors named centroids or centers. But now the clusters need the centroids plus w vectors and diameters d .

Definition. Given a set S of N points \mathbf{x} in the D -dimensional Euclidean space, a set of k centers $\{\mathbf{c}_1, \dots, \mathbf{c}_k\}$, $\mathbf{c}_j \in D$, $j = 1, \dots, k$, coupled with a set of corresponding weight vectors $\{\mathbf{w}_1, \dots, \mathbf{w}_k\}$, $\mathbf{w}_j \in D$, $j = 1, \dots, k$, and also a set of diameters d_j , $\mathbf{d} \in [0,1]^k$, partition S into k sets $\{S_1, \dots, S_k\}$:

$$S_j = \left\{ x \mid \sum_{i=1}^D d_j (w_{ji} (x_i - c_{ji})^2) < \sum_{i=1}^D d_l (w_{li} (x_i - c_{li})^2) \forall l \neq j \right\} \quad 1$$

where w_{ji} represents the i th components of vectors \mathbf{w}_j as well as c_{ji} is i th dimension of \mathbf{c}_j , respectively, finally d_j stands for diameter of cluster j (ties are broken randomly).

The set of centers and weights are optimal with respect to the Euclidean norm, if they minimize the error measure:

$$E_1(C, D, W) = \sum_{j=1}^k \sum_{i=1}^D d_j \left(w_{ji} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right) \quad 2$$

subject to the constraints $\forall j \sum_i w_{ji} = 1$ and $\sum_j d_j = 1$. C and W are $(D \times k)$ matrices whose column vectors are \mathbf{c}_j and \mathbf{w}_j , respectively, i.e., $C = [\mathbf{c}_1 \dots \mathbf{c}_k]$ and $W = [\mathbf{w}_1 \dots \mathbf{w}_k]$, and $|S_j|$ is the cardinality of set S_j . Solving the equation below, the vector \mathbf{D} will be non-zero for all clusters and zero for one cluster and all data points will be assigned to the cluster corresponds to zero in \mathbf{D} vector.

$$(C_*, D_*, W_*) = \arg \min_{(C, D, W)} E_1(C, D, W) \quad 3$$

Our objective, instead, is to find diametric clusters partitioning, where the unit weight gets distributed among all clusters according to the respective variance of data within each cluster. One way to achieve this goal is to add the regularization term $\sum_{j=1}^k d_j \log d_j$ which represents the negative entropy of the diameter distribution for clusters [8]. It penalizes solutions with minimal (zero) diameter on the single cluster. The resulting error function will be as follow:

$$E_2(C, D, W) = \sum_{j=1}^k \sum_{i=1}^D d_j \left(w_{ji} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right) + h_1 \sum_{j=1}^k d_j \log d_j \quad 4$$

subject to the constraint $\sum_j d_j = 1$, where h_1 is parameter of criterion error. But now as it is said the solution of above error criterion will results in maximal (i.e., unit) weight on the feature with smallest variance in each clusters [4]. So, we add the regularization term $\sum_{i=1}^D w_{ij} \log w_{ij}$ to our proposed error criterion. Then the proposed error criterion will be as follow:

$$E_3(C, D, W) = \sum_{j=1}^k \sum_{i=1}^D \left(d_j w_{ji} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 + h_2 \sum_{i=1}^D w_{ij} \log w_{ij} \right) + h_1 \sum_{j=1}^k d_j \log d_j \quad 5$$

subject to the constraints $\sum_j w_{ji} = 1$ and $\sum_j d_j = 1$. The coefficient $h_1, h_2 \geq 0$ are the parameters of the procedure. Parameters h_1 and h_2 control how much the distribution of weight values will deviate from the uniform distribution. We can solve this constrained optimization problem by introducing the Lagrange multipliers λ_j (one for each constraint) and μ for $\sum_j d_j = 1$, and minimizing the final (unconstrained now) error criterion:

$$\begin{aligned} E(C, D, W) &= \sum_{j=1}^k \sum_{i=1}^D \left(d_j w_{ji} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 + h_2 \sum_{i=1}^D w_{ij} \log w_{ij} \right) \\ &\quad + h_1 \sum_{j=1}^k d_j \log d_j + \sum_{j=1}^k \lambda_j (1 - \sum_{i=1}^D w_{ij}) + \mu (1 - \sum_{j=1}^k d_j) \end{aligned} \quad 6$$

For a fixed partition P , fixed c_{ji} and fixed d_j , we compute the optimal w_{ij} by setting $\frac{\partial E}{\partial w_{ji}} = 0$ and $\frac{\partial E}{\partial \lambda_j} = 0$. We obtain:

$$\frac{\partial E}{\partial w_{ji}} = d_j \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 + h_2 \log(w_{ji}) + h_2 - \lambda_j = 0 \quad 7$$

$$\frac{\partial E}{\partial \lambda_j} = 1 - \sum_{i=1}^D w_{ij} = 0 \quad 8$$

solving equation 7 with respect to w_{ji} we obtain

$$h_2 \log(w_{ji}) = -d_j \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 - h_2 + \lambda_j \quad 9$$

$$w_{ji} = \exp(-\frac{d_j}{h_2} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2) / \exp(1 - \frac{\lambda_j}{h_2}) \quad 10$$

substituting this expression in equation 8 yields to:

$$\frac{\partial E}{\partial \lambda_j} = 1 - \exp(\frac{\lambda_j}{h_2}) \sum_{i=1}^D \exp(-\frac{d_j}{h_2} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 - 1) = 0 \quad 11$$

solving with respect to λ_j we obtain

$$\lambda_j = -h_2 \log(\sum_{i=1}^D \exp(-\frac{d_j}{h_2} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 - 1)) = 0 \quad 12$$

solving w_{ji}^* with considering w_{ji} and λ_j yields to:

$$w_{ji}^* = \frac{\exp\left(-\frac{d_j}{h_2} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2\right)}{\sum_{i=1}^D \exp\left(-\frac{d_j}{h_2} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2\right)} \quad 13$$

For a fixed partition P , fixed c_{ji} and fixed w_{ji} , we compute the optimal d_j by setting $\frac{\partial E}{\partial d_j} = 0$ and $\frac{\partial E}{\partial \mu} = 0$. We obtain:

$$\frac{\partial E}{\partial d_j} = \sum_{i=1}^D \left(w_{ji} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right) + h_1 \log(d_j) + h_1 - \mu = 0 \quad 14$$

$$\frac{\partial E}{\partial \mu} = (1 - \sum_{j=1}^k d_j) = 0 \quad 15$$

again, solving first equation with respect to d_j we obtain

$$h_1 \log(d_j) = - \sum_{i=1}^D \left(w_{ji} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right) - h_1 + \mu \quad 16$$

$$d_j = \frac{\exp\left(-\sum_{i=1}^D \left(\frac{w_{ji}}{h_1} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right)\right)}{\exp(1 - \frac{\mu}{h_1})} \quad 17$$

substituting this expression in equation 15 yields to:

$$1 - \sum_{j=1}^k \frac{\exp\left(-\sum_{i=1}^D \left(\frac{w_{ji}}{h_1} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right)\right)}{\exp(1 - \frac{\mu}{h_1})} = 1 - \exp\left(\frac{\mu}{h_1}\right) \sum_{j=1}^k \exp\left(-\sum_{i=1}^D \left(\frac{w_{ji}}{h_1} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right)\right) - 1 = 0 \quad 18$$

solving with respect to μ we obtain

$$\mu = -h_1 \log\left(\sum_{j=1}^k \exp\left(-\sum_{i=1}^D \left(\frac{w_{ji}}{h_1} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2 \right)\right) - 1\right) = 0 \quad 19$$

solving d_j^* with considering d_j and μ yields to:

$$d = \frac{\exp\left(-\sum_{i=1}^D \left(\frac{w_{ji}}{h_1} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2\right)\right)}{\sum_{j=1}^k \exp\left(-\sum_{i=1}^D \left(\frac{w_{ji}}{h_1} \frac{1}{|S_j|} \sum_{x \in S_j} (x_i - c_{ji})^2\right)\right)} \quad 20$$

For a fixed partition P and fixed w_{ji} , we compute the optimal c_{ji}^* by setting $\frac{\partial E}{\partial c_{ji}} = 0$.

We obtain:

$$\frac{\partial E}{\partial c_{ji}} = \frac{2d_j w_{ji}}{|S_j|} \sum_{x \in S_j} (c_{ji} - x_i) = \frac{2d_j w_{ji}}{|S_j|} \left(|S_j| c_{ji} - \sum_{x \in S_j} x_i \right) = 0 \quad 21$$

Solving with respect to c_{ji} gives

$$c_{ji}^* = \frac{1}{|S_j|} \sum_{x \in S_j} x_i \quad 22$$

Proposition. When $h_1 = 0$ and $h_2 = 0$, the error function E3 is reduced to E1; when $h_1 = \infty$ and $h_2 = \infty$, the error function E3 is reduced to SSE.

4 Search Strategy

We need to provide a search strategy to find a partition P that identifies the final partition. Our approach is similar to k-means and LAC where they iteratively improve the quality of initial centroids, weights and diameters, by investigating the space near the centers to estimate the dimensions that matter the most. Specifically, we proceed as follows. We start with *well-scattered* points in S as the k centroids: we choose the first centroid at random, and select iteratively the others so that they are as far as possible from each so far-selected centroids. We initially set all weights and all diameters to \bar{ID} and $1/k$ respectively. Given the initial centroids c_j , for $j = 1, \dots, k$, we compute the corresponding sets S_j as given in the definition above. Then we use the S_j to compute weights W_j . The computed weights are used to compute the diameters D_j , and finally S_j , D_j and W_j are used to update centroids c_j .

Input. N points $\mathbf{x} \in R^D$, k , and h_1 , h_2 .

1. Start with k initial centroids c_1, c_2, \dots, c_k ;
2. Set $d_j = \bar{1}k$, for each centroid c_j , $j = 1, \dots, k$
3. Set $w_{ji} = \bar{1}D$, for each centroid c_j , $j = 1, \dots, k$ and each feature $i = 1, \dots, D$
4. For each centroid c_j , compute S_j considering w_{ji} and d_j ;
5. Compute new weights w_{ji} using c_j and d_j .
6. For each centroid c_j , compute d_j considering w_{ji} and S_j ;
7. Compute new centroids.
8. Iterate 4,5,6,7 until convergence.

Table 1. Experimental results. * indicates dataset is normalized with mean of 0 and variance of 1, $N(0,1)$.

Dataset	Simple Methods (%)					
	Single Linkage	Average Linkage	Complete Linkage	K-means	Fuzzy K-means	Proposed Algorithm
Breast Cancer*	65.15	65.15	65.15	95.37	55.34	98.42
Bupa*	58.26	57.68	57.68	54.49	37.77	59.36
Glass*	35.05	37.85	37.85	45.14	49.07	53.41
Wine	38.76	37.64	39.89	96.63	96.63	98.11
Yeast*	34.38	35.11	38.91	40.20	35.65	47.34
Iris	66.67	67.33	38.00	82.80	89.33	94.33
SAHeart*	65.15	65.37	64.72	63.12	45.19	69.03
Ionosphere*	63.82	67.52	65.81	70.66	53.22	72.84
Galaxy*	25.70	25.70	25.70	29.88	29.41	36.19

5 Experimental Results

This section evaluates the result of applying proposed algorithm on 9 real datasets available at USI repository [15]. The final performance of the clustering algorithm is evaluated by re-labeling between obtained partition and the ground true labels and then counting the percentage of the true classified samples. Table 1 shows the performance of the proposed method comparing with the most common base methods. To reach these results for proposed clustering algorithm, by trial and error we turn to the best parameter values for h_1 and h_2 per each dataset. As it can be seen in Fig. 2 the parameters h_1 and h_2 are set for all dataset in the ranges [0.06,0.15] and [0.09,0.27]. These ranges are obtained by trial and error to be the best options for all datasets.

The four first columns of Table 1 are the results of some base clustering algorithms. The results show that although each of these algorithms can obtain a good result over a specific dataset, it does not perform well over other datasets. But well setting of parameters of proposed method leads to its perfect superiority to most of well-known clustering algorithms. The only drawback of the proposed algorithm is the sensitivity to its two parameters. For a comprehensive study, look at the Fig 2 and Fig. 3. In Fig. 2, the normalized mutual information (*NMI*) values between output labels of WLAC algorithm and real labels for different values of parameters h_1 and h_2 over Iris dataset are depicted. It is worthy to mention that the diagram is a cut from its best positions over all possible values for its parameters (h_1 and h_2 both are defined in the range $[0, \infty]$). In Fig. 3, the normalized mutual information (*NMI*) values between output labels of LAC algorithm and real labels for different values of parameters h over Iris dataset are depicted. Note that the diagram is again a cut from its best positions over all possible values for its parameter (h is defined in the range $[0, \infty]$, i.e. y is defined in range $[1, \infty]$ and x is defined in range $[1, 10]$). For a more detailed view of Fig. 3, look at the Fig. 4. It is obvious that in the best setting of parameter for LAC algorithm in the Iris dataset, it can't outperform WLAC algorithm with just a near best setting of its parameters, i.e. as it is presented in Fig. 2. Also note that WLAC is less sensitive to its well setting of parameters than LAC to its well setting of parameter.

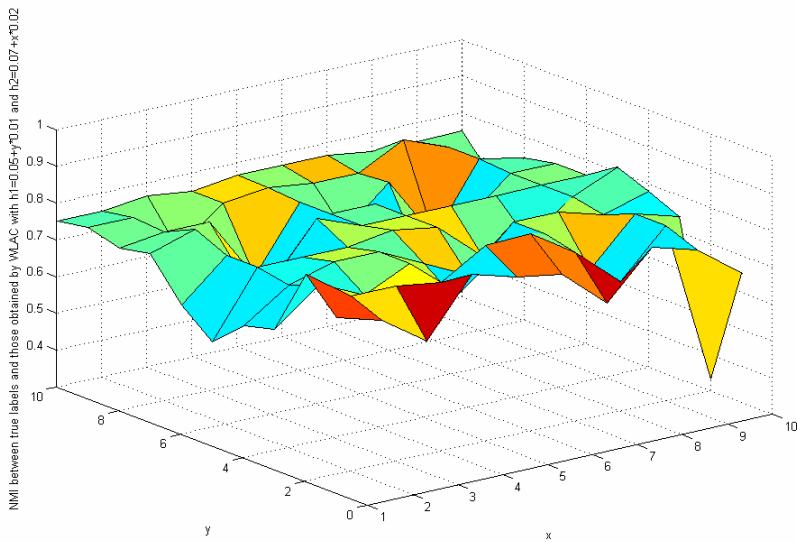


Fig. 2. NMI between real labels of Iris dataset and those of obtained by WLAC algorithm using different values of parameters h_1 and h_2

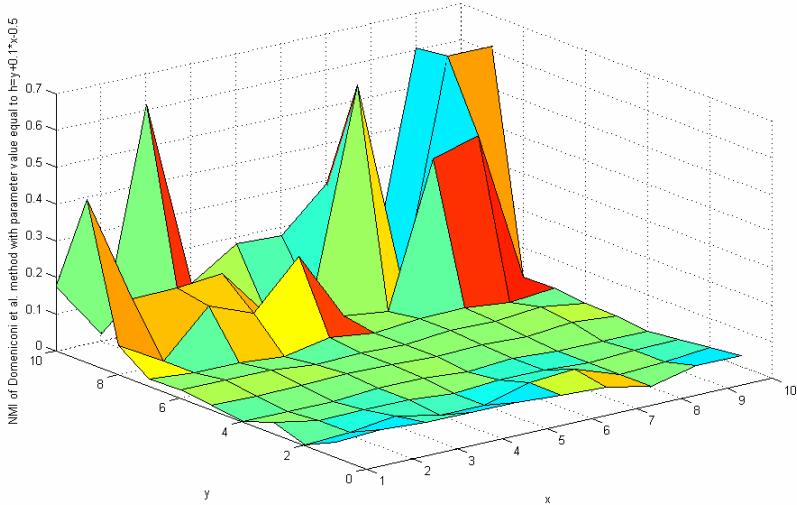


Fig. 3. NMI between real labels of Iris dataset and those of obtained by LAC algorithm using different values of parameters h ($h=y+0.1*x-0.5$)

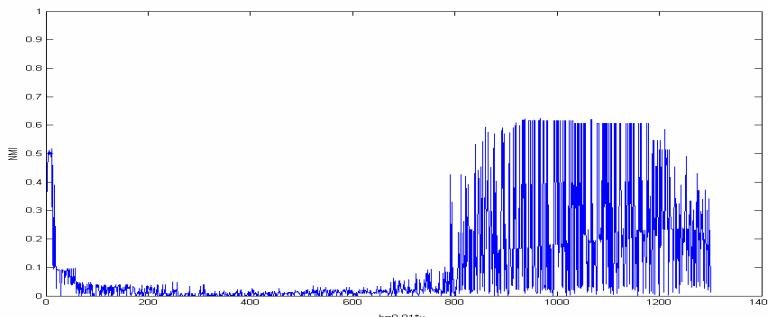


Fig. 4. NMI between real labels of Iris dataset and those of obtained by LAC algorithm using different values of parameters h ($h=0.01*x$)

6 Conclusion and Future Work

This paper proposes a new metric in clustering which simultaneously considers the feature weighting and cluster weighting. It is also solved in algebraic mathematic so as to obtain the minima. A new algorithm based on k-means is presented to handle the amenities added to k-means metric, i.e. Sum of Square Errors (SSE). The proposed method have two parameters which must be appropriately set to obtain a well output partitioning. Tuning these parameters can be an open problem as future work which we are working on it.

The only drawback of the proposed algorithm is its sensitivity to its two parameters. But well setting of its parameters leads to its perfect superiority to most of well-known clustering algorithms. So for future work, it can be studied how to overcome the problem of automatic setting of its two parameters.

References

1. Blum, A., Rivest, R.: Training a 3-node neural networks is NP-complete. *Neural Networks* 5, 117–127 (1992)
2. Chang, J.W., Jin, D.S.: A new cell-based clustering method for large, high-dimensional data in data mining applications. In: Proceedings of the ACM Symposium on Applied Computing, pp. 503–507. ACM Press, New York (2002)
3. Cheng, C.H., Fu, A.W., Zhang, Y.: Entropy-based subspace clustering for mining numerical data. In: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 84–93. ACM Press, New York (1999)
4. Domeniconi, C., Al-Razgan, M.: Weighted cluster ensembles: Methods and analysis. *TKDD* 2(4) (2009)
5. Domeniconi, C., Gunopulos, D., Ma, S., Yan, B., Al-Razgan, M., Papadopoulos, D.: Locally adaptive metrics for clustering high dimensional data. *Data Min. Knowl. Discov.* 14(1), 63–97 (2007)

6. Dudoit, S., Fridlyand, J.: Bagging to improve the accuracy of a clustering procedure. *Bioinformatics* 19(9), 1090–1099 (2003)
7. Faceli, K., Marcilio, C.P., Souto, d.: Multi-objective Clustering Ensemble. In: Proceedings of the Sixth International Conference on Hybrid Intelligent Systems (HIS 2006) (2006)
8. Friedman, J.H., Meulman, J.J.: Clustering objects on subsets of variables. *Journal of the Royal Statistical Society, Series B* 66, 815–849 (2004)
9. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice Hall, Englewood Cliffs (1988)
10. Kohavi, R., John, G.: Wrappers for feature subset selection. *Artificial Intelligence* 97(1-2), 273–324 (1997)
11. Liu, B., Xia, Y., Yu, P.S.: Clustering through decision tree construction. In: Proceedings of the Ninth International Conference on Information and Knowledge Management, pp. 20–29. ACM Press, New York (2000)
12. MacQueen, J.B.: Some Methods for classification and Analysis of Multivariate Observations. In: 5th Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297. University of California Press, Berkeley (1967)
13. Miller, R., Yang, Y.: Association rules over interval data. In: Proc. ACM SIGMOD International Conf. on Management of Data, pp. 452–461 (1997)
14. Mirzaei, A., Rahmati, M., Ahmadi, M.: A new method for hierarchical clustering combination. *Intelligent Data Analysis* 12(6), 549–571 (2008)
15. Newman, C.B.D.J., Hettich, S., Merz, C.: UCI repository of machine learning databases (1998), <http://www.ics.uci.edu/~mlearn/MLSummary.html>
16. Parsons, L., Haque, E., Liu, H.: Subspace clustering for high dimensional data: a review. *ACM SIGKDD Explorations Newsletter* 6(1), 90–105 (2004)
17. Srikant, R., Agrawal, R.: Mining Quantitative Association Rules in Large Relational Tables. In: Proc. of the ACM SIGMOD Conference on Management of Data, Montreal, Canada (1996)

On the Comparison of Parallel Island-Based Models for the Multiobjectivised Antenna Positioning Problem

Eduardo Segredo, Carlos Segura, and Coromoto León

Dpto. Estadística, I.O.y Computación. Universidad de La Laguna,
La Laguna, 38271, Santa Cruz de Tenerife, Spain
`{esegredo,csegura,cleon}@ull.es`

Abstract. Antenna Positioning Problem (APP) is an NP-Complete Optimisation Problem which arises in the telecommunication field. Its aim is to identify the infrastructures required to establish a wireless network. A well-known mono-objective version of the problem has been used. The best-known approach to tackle such a version is a problem-dependent strategy. However, other methods which minimise the usage of problem-dependent information have also been defined. Specifically, *multiobjectivisation* has provided solutions of similar quality than problem-dependent strategies. However, it requires a larger amount of time to converge to high-quality solutions. The main aim of the present work has been the decrease of the time invested in solving APP with multi-objectivisation. For this purpose, a parallel island-based model has been applied to two APP instances. In order to check the robustness of the approach, several migration stages have been tested. In addition, a scalability analysis using the best-behaved migration stage has been performed. Computational results have demonstrated the validity of the proposal.

1 Introduction

The Antenna Positioning Problem (APP) and the Assignment Frequency Problem (AFP) are the major problems [10] which arise in the engineering of mobile telecommunication networks. The APP identifies the most promising sites to locate a set of Base Stations (BS) or antennas. The sites are selected among a list of potential ones. Several objectives can be taken into account for this purpose. Most typical considered objectives are minimising the number of antennas, maximising the amount of traffic held by the network, maximising the quality of service, and/or maximising the covered area. In addition, a set of constraints may be considered. The AFP sets up frequencies used by such antennas with the aim of minimising interferences, i.e., maximising the offered quality of service. Both problems play a major role in various engineering, industrial, and scientific applications because its outcome usually affects cost, profit, and other heavy-impact business performance metrics. This means that the quality of the applied approaches has a direct bearing on industry economic plans.

In this paper, the APP has been addressed. The APP has been referred in the literature using alternative names, mainly: *Radio Network Design* (RND) and *Base Station Transmitters Location Problem* (BST-L). The APP is a very complex problem. In fact, it has been demonstrated to be a NP-complete problem [8]. The APP and the AFP have been jointly analysed in some cases [1]. In other cases, they have been considered as independent problems [9]. Several formulations of the APP have been proposed [11]. Most of them have been mono-objective proposals [9]. In [14], the APP was tackled as a mono-objective problem by translating the other considered objectives into restrictions. In [13], several objectives were considered simultaneously and multi-objective strategies were applied. In this paper, the mono-objective variant presented in [2][6] has been used. In this version, the fitness function takes into consideration the coverage of the deployed network and the number of used BS.

Many strategies have been applied to the mono-objective and multi-objective versions of the APP. Most of them incorporate problem-dependent information. Therefore, adapting them to other variants of the problem is not a trivial task. Moreover, such approaches have a huge cost associated to its design. In [1][4] completely ad-hoc heuristics were designed. Evolutionary strategies were applied in [2][6]. In [16], problem-dependent information was included in the mutation operators. A wide comparison of mono-objective techniques applied to the here considered version of the APP was performed in [8]. Problem-independent techniques achieved poorer quality solutions than those which incorporated problem-dependent information. However, given the drawbacks of problem-dependent approaches, several alternatives that minimises the usage of problem-dependent information have also been tested. In [11], Multi-objective Evolutionary Algorithms (MOEAs) were applied. The here considered mathematical formulation was used, but the coverage and the number of BS were considered as two separate objectives. The diversity of the solutions was improved. However, the MOEAs were not able to achieve as high fitness values as problem-dependent approaches. Another alternative resides on the usage of multiobjectivisation. The term *multiobjectivisation* was introduced in [7] to refer to the reformulation of originally mono-objective problems as multi-objective ones. In [12], multiobjectivised approaches had a slower convergence than problem-dependent techniques. However, in the long term they were able to achieve solutions of similar quality.

In order to reduce the computational time, several studies have considered the parallelisation of MOEAs [3]. Parallel Multi-Objective Evolutionary Algorithms (pMOEAs) can be classified [5] in three major computational paradigms: *master-slave*, *island-based*, and *diffusion*. When compared to other parallel models, the island-based approach brings two benefits: it maps easily onto the parallel architectures, and it extends the search area, trying to avoid local optima stagnation. Island-based models have shown good performance and scalability in many areas [3]. They conceptually divide the overall pMOEA population into a number of independent populations, i.e., there are separate and simultaneously MOEAs executing on each island. Each island evolves in isolation for the majority of the pMOEA execution, but occasionally, some individuals can be migrated between

neighbour islands. Given that the migration stage allows the collaboration among islands, it is an essential operation on these parallel schemes. A well-designed migration stage could provide a successful collaboration.

In this paper, the validity of the hybridisation among parallel island-based models and multiobjectivised strategies applied to the APP has been tested. A comparison among several migration stages has been carried out to check the robustness of the approach. Moreover, a scalability study with the best-behaved migration stage has been performed. The main aim has been the decrease of the time invested by multiobjectivised strategies to achieve a similar quality level than the one obtained by problem-dependent approaches.

The remaining content of the work is structured in the following way: the mathematical formulation of the problem is given in Sect. 2. Section 3 is devoted to describe the applied optimisation scheme. The sequential strategy is detailed in Sect. 3.1. Specifically, the multiobjectivised methods and genetic operators are described. In Sect. 3.2 the applied parallel island-based model is depicted. Then, the experimental evaluation is presented in Sect. 4. Finally, the conclusions and some lines of future work are given in Sect. 5.

2 APP: Mathematical Formulation

The APP is defined as the problem of identifying the infrastructures required to establish a wireless network. It comprises the maximisation of the coverage of a given geographical area while minimising the BS deployment. A BS is a radio signal-transmitting device that irradiates any type of wave model. The region of the area covered by a BS is called a cell. In the considered APP definition, a BS can only be located in a set of potential locations. The APP mathematical formulation here used was presented in [216]. In this formulation, the fitness function is defined as:

$$f(\text{solution}) = \frac{\text{Cover}^\alpha}{\text{Transmitters}}$$

In the previous scheme, a decision maker must select a value for α . It is tuned considering the importance given to the coverage, in relation with the number of deployed BS. As in [216], the parameter $\alpha = 2$ has been used.

The geographical area G on which a network is deployed is discretised into a finite number of points or locations. Tam_x and Tam_y are the number of vertical and horizontal subdivisions, respectively. They are selected by communications experts, depending on several characteristics of the region and transmitters. U is the set of locations where BS can be deployed: $U = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$. Location i is referred using the notation $U[i]$. The x and y coordinates of location i are named $U[i]_x$ and $U[i]_y$, respectively. When a BS is located in position i , its corresponding cell is covered. The cell is named $C[i]$. In this work, the APP canonical formulation has been used, i.e., an isotropic radiating model has been considered for the cell. The set P determines the locations covered by a BS: $P = \{(\Delta x_1, \Delta y_1), (\Delta x_2, \Delta y_2), \dots, (\Delta x_m, \Delta y_m)\}$. Thus, if BS i is deployed, the covered locations are given by the next set: $C[i] = \{(U[i]_x +$

$\Delta x_1, U[i]_y + \Delta y_1), (U[i]_x + \Delta x_2, U[i]_y + \Delta y_2), \dots, (U[i]_x + \Delta x_m, U[i]_y + \Delta y_m)\}$. Being $B = [b_0, b_1, \dots, b_n]$ the binary vector which determines the deployed BS, the next definitions holds for APP:

$$\text{Transmitters} = \sum_{i=0}^n b_i \quad \text{Cover} = \frac{\sum_{i=0}^{tam_x} \sum_{j=0}^{tam_y} covered(i,j)}{tam_x \times tam_y} \times 100$$

where:

$$covered(x, y) = \begin{cases} 1 & \text{If } \exists i / \{(b_i = 1) \wedge ((x, y) \in C[i])\} \\ 0 & \text{Otherwise} \end{cases}$$

3 Optimisation Scheme

3.1 Sequential Strategy

The best-behaved optimisation scheme among the proposed in [12] has been used in the current work. It is based in the well-known *Non-dominated Sorting Genetic Algorithm II* (NSGA-II). In that approach, an artificial objective function was added to multiobjectivise the APP. Multiobjectivisation changes the fitness landscape, so it can be useful to avoid local optima [6], and consequently, to make easier the resolution of the problem. However, it can also produce a harder problem [4]. There are two different ways of multiobjectivising a problem. The first one is based on a decomposition of the original objective, while the second one is based on aggregating new objective functions. The aggregation of alternative functions can be performed by considering problem-dependent or problem-independent information. To multiobjectivise the APP, the first objective was selected as the original fitness function, while for the second one, an artificial function which tries to maximise the diversity was used. Several artificial functions were tested in [12]. The one which obtained the best results is based on using the Euclidean distance to the best individual in the population. Since selection pressure is decreased, some low quality individuals could be maintained along several generations. However, these individuals could help to avoid stagnation in local optima. In fact, in [12] the best-behaved multiobjectivised approach was able to achieve, in the long term, solutions of similar quality than problem-dependent strategies.

The NSGA-II makes use of a variation phase, which consists in the application of mutation and crossover operators. The best-behaved genetic operators applied in [11] have been considered. The applied mutation operator has been the well-known *Bit Inversion Mutation*. Each gene is inverted with a probability p_m . The used crossover operator has been the *Geographic Crossover*. It exchanges the BS which are located within a given radius (r) around a randomly chosen BS. It is applied with a probability p_c . Tentative solutions have been represented as binary strings with n elements, where n is the number of potential BS. Each gene determines whether the corresponding BS is deployed.

3.2 Parallel Strategy

In order to reduce the execution time required to attain high quality solutions with the aforementioned sequential approach, parallelisation has been considered. Specifically, an island-based model has been applied. In island-based models, the population is divided into a number of independent subpopulations. Each subpopulation is associated to an island and a MOEA configuration is executed over each subpopulation. Usually, each available processor constitutes an island which evolves in isolation for the majority of the parallel run. However, collaborative schemes could lead to a better behaviour. Therefore, a migration stage which enables the transfer of individuals among islands is generally incorporated.

Four basic island-based models are seen to exist [5]: all islands execute identical configurations (homogeneous), all islands execute different configurations (heterogeneous), each island evaluates different objective function subsets, and each island represents a different region of the genotype or phenotype domains. In the first two variants, the population of each island represents solutions to the same problem. In the third variant, each island searches a reduced problem domain space. The last variant isolates each processor to solve specific, non-overlapping regions of genotype/phenotype domain space. The parallel strategy presented in this paper is based on the homogeneous island-based model. Each island of this parallel strategy executes the approach exposed in Sect. 3.1.

Given that the migration stage allows the collaboration among islands, it is an essential operation on these parallel schemes. A well designed migration stage could provide a successful collaboration. Thus, the solution search space could be better explored and higher quality solutions could be obtained. However, if a not suitable migration stage is applied inside the model, the effect could be similar, or even worse, than having separate MOEAs simultaneously executing on several processors with no communication among them. Therefore, the migration stage must be carefully defined. In order to configure the migration stage, it is necessary to establish the migration topology (where to migrate the individuals) and the migration rate (the maximum number of individuals to be migrated at each step and how often the migration stage is executed). In addition, individuals which are going to be migrated and those which are going to be replaced must be selected. Such a selection is performed by the use of the migration scheme and the replacement scheme, respectively.

Fitness landscapes of island-based models may be completely different from those produced by their corresponding sequential MOEAs. As such the pMOEA may find better or equivalent solutions quicker or even slower. Depending on the selected migration stage, the landscape is affected on different ways [15]. Therefore, four different migration stages have been tested in this work. The migration stages [15] have been constituted by combining different migration and replacement schemes. Two migration schemes have been tested: *Elitist* (ELI), and *Random* (RND). In ELI, a subpopulation individual is selected if it is better than any member of its previous generation. In the RND scheme, subpopulation individuals are randomly selected. Also, two replacement schemes have been

analysed: *Elitist Ranking* (ELI), and *Random* (RND). In the ELI scheme, first, the subpopulation individuals of the destination island are ranked following the NSGA-II *Crowding Operator*. Then, individuals randomly selected from the worst available rank are replaced. In the RND replacement scheme, subpopulation individuals are randomly selected. Each migration stage has been identified with the following acronym: *migration-replacement*. For example, ELI-RND means that an elitist migration scheme has been combined with a random replacement scheme.

4 Experimental Evaluation

In this section the experiments performed with the different optimisation schemes depicted in Sect. 3 are described. Tests have been run on a Debian GNU/Linux computer with 4 AMD ® Opteron™ (model number 6164 HE) at 1.7 GHz and 64 Gb RAM. The compiler which has been used is *gcc 4.4.5*. Two APP instances have been analysed. The first one is a real world-sized problem instance. It is defined by the geographical layout of the city of Málaga (Spain). This instance represents an urban area of 27.2 km^2 . The terrain has been modelled using a 450 x 300 grid, where each point represents a surface of approximately 15 x 15 m. The dataset contains $n = 1000$ candidate sites for the BS. The second instance is an artificial generated one. In this case, the terrain has been modelled using a 287 x 287 grid. The dataset contains $n = 349$ candidate sites for the BS.

Since experiments have involved the use of stochastic algorithms, each execution has been repeated 30 times. Each experiment has been carried out for both instances. In order to provide the results with statistical confidence, comparisons have been performed applying the following statistical analysis. First, a *Shapiro-Wilk test* is performed in order to check whether the values of the results follow a normal (Gaussian) distribution or not. If so, the *Levene test* checks for the homogeneity of the variances. If samples have equal variance, an *ANOVA test* is done. Otherwise, a *Welch test* is performed. For non-Gaussian distributions, the non-parametric *Kruskal-Wallis test* is used to compare the medians of the algorithms. A confidence level of 95% is considered, which means that the differences are unlikely to have occurred by chance with a probability of 95%.

For all experiments, the following parameterisation has been used: $r = 30$, $p_c = 1$, and $p_m = \frac{1}{n}$, being n the number of potential sites. For the artificial instance the population size has been fixed to 50 individuals, and for the Málaga instance a population size of 100 individuals has been used.

In the first experiment, a robustness analysis of the parallel model in terms of the applied migration stage has been carried out. The parallel island-based model has been tested with the 4 migration stages described in Sect. 3.2. They have been executed with 4 islands, and with a stopping criterion of 6 hours. In every case, an all to all connected migration topology has been applied. The migration rate has been fixed to 1 individual and a migration probability equal to 0.01 has been used. Figure 11 shows the evolution of the average fitness values for the parallel and sequential approaches. For both instances, the parallel approaches have clearly improved the results obtained by the sequential strategy.

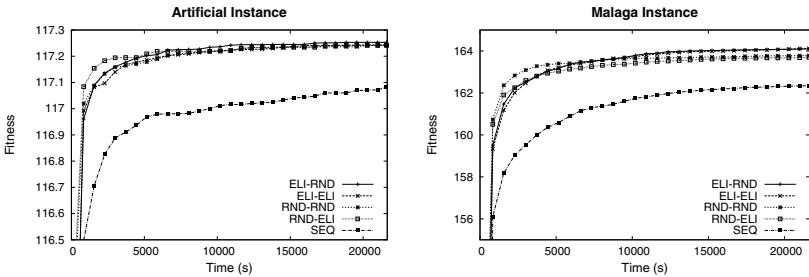


Fig. 1. Fitness evolution of the parallel island-based model with 4 islands

All parallel island-based models have obtained similar fitness values. In fact, the statistical analysis has revealed that differences among them has not been significant. In both cases, the highest average fitness value has been obtained by the parallel model which has used the ELI-RND migration stage. Since high quality results have been obtained by every parallel model regardless of the applied migration scheme, the robustness of the proposal has been demonstrated.

Given that the parallel models have used more computational resources than the sequential model, the obtained improvement must be quantified. *Run-length distributions* are useful to quantify the improvement of the parallel models. They show the relation between success ratios and time. The success ratio is defined as the probability of achieving a certain quality level. Run-length distributions have been calculated for the parallel island-based models, and for the sequential strategy. In the case of the artificial instance, since each parallel scheme has been able to achieve the best currently known fitness value, such a value has been selected as the quality level. For the Málaga instance, the variance of the results has been higher than for the artificial instance. Thus, if the best currently known fitness value had been selected as the quality level, very low success ratios would have been achieved. Therefore, the quality level has been selected so that at least a 60% of success ratio has been achieved by every parallel island-based model. Figure 2 shows the run-length distributions of the parallel and sequential models for both instances. In the case of the sequential model, a maximum execution time of 24 hours has been considered. For the parallel models, the maximum considered execution time has been 6 hours. Run-length distributions have confirmed the superiority of the parallel approaches. In fact, superlinear speedup factors have been obtained. The main reason has been the ability of the parallel models to extend the search space, and consequently, to deal with local optima stagnation. Run-length distributions have also shown the similarities among the different parallel approaches. However, when high success ratios have been taken into account, ELI-RND has been the best-behaved strategy.

The second experiment has analysed the scalability of the proposed parallel strategy. The parallel island-based model with the best-behaved migration stage (ELI-RND, referenced in this second experiment as PAR4) has been executed with 8 islands (PAR8) and 16 islands (PAR16). Figure 3 shows their run-length distributions considering a maximum execution time of 6 hours for both instances.

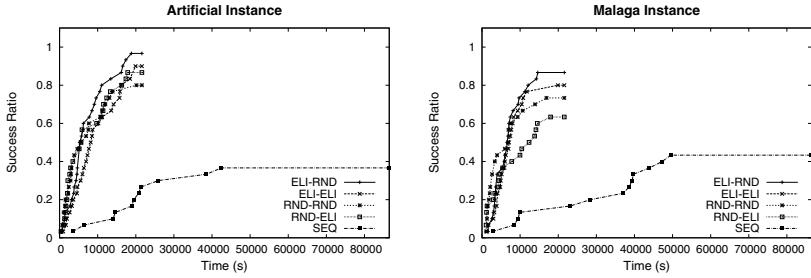


Fig. 2. Run-length distributions of the parallel island-based models with 4 islands

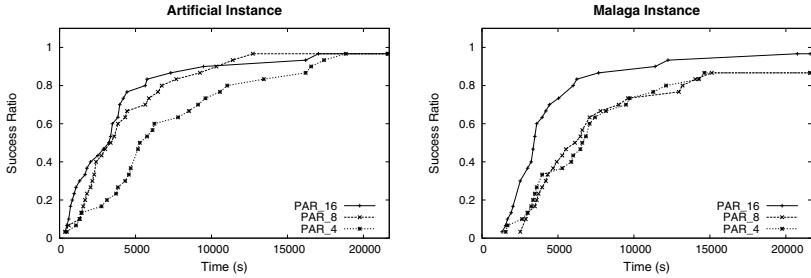


Fig. 3. Run-length distributions of the parallel models (4, 8, and 16 islands)

They show the benefits of incorporating additional islands. Speedup factors, taking as reference PAR4, have been calculated for different success ratios, ranging from a 25% to a 75%. In the case of the artificial instance, the speedup factor of PAR8 has ranged from 1.57 to 1.88. For the model PAR16, they have ranged from 1.62 to 3.57. Some punctual scalability problems have been detected for PAR16. In fact, PAR8 and PAR16 have provided similar speedup factors for some success ratios. However, PAR16 has obtained higher speedup factors for other success ratios, showing its benefits. For the Málaga instance, some scalability problems have also been detected. The benefits obtained by PAR8 have been negligible. In fact, the run-length distributions of PAR4 and PAR8 have been similar. However, when a higher number of islands has been considered (PAR16), the obtained speedup factors have increased. The speedup factors of PAR16, taking as reference PAR4, have ranged from 1.42 to 1.9.

5 Conclusions and Future Work

The APP is one of the major problems which arises in the engineering of mobile telecommunication networks. In this paper, an analysis of the hybridisation among parallel island-based models and multiobjectivised strategies applied to the APP has been performed. This analysis has been carried out with two different instances. Multiobjectivisation is a more general strategy than those which incorporate problem-dependent information. In [12], several multiobjectivisations of

the APP were proposed. The optimisation scheme was based on the NSGA-II. The best-behaved multiobjectivised method took into account the Euclidean distance to the best individual of the population. The main drawback of this approach was the increase of the required time to attain high quality solutions, when multiobjectivisation was compared with problem-dependent strategies. In order to decrease the convergence time, a homogeneous island-based model has been applied. The configuration executed on the islands has been the best-behaved multiobjectivised approach proposed in [2]. Migrations are an essential operation in these parallel schemes. Therefore, a robustness analysis of the parallel model has been performed taking into consideration different migration stages. The experimental evaluation has demonstrated the robustness of the parallel approach regardless of the incorporated migration stage. In addition, every parallel approach have clearly improved the results obtained by the corresponding sequential approach. In fact, superlinear speedup factors, have been obtained by them, when 4 islands have been used. Afterwards, a scalability analysis up to 16 islands with the best-behaved migration stage (ELI-RND) has been performed. For both instances, some scalability problems have been detected. The time invested in achieving high quality solutions has been decreased by incorporating additional processors. However, the decrease has not always been in a linear way.

Future work will be focused in the application of parallel hyperheuristics to solve the APP. Since the appropriate optimisation method could depend on the instance that is being solved, the application of hyperheuristics seems a promising approach. Thus, the selection of the optimisation method which is used on each island, could be performed in an automatic way. In addition, it would be interesting to analyse other APP instances.

Acknowledgements. This work was supported by the EC (FEDER) and the Spanish Ministry of Science and Innovation as part of the 'Plan Nacional de I+D+i', with contract number TIN2008-06491-C04-02 and by Canary Government project PI2007/015. The work of Eduardo Segredo and Carlos Segura was funded by grants FPU-AP2009-0457 and FPU-AP2008-03213, respectively. The work was also funded by the HPC-EUROPA2 project (project number: 228398) with the support of the European Commission - Capacities Area - Research Infrastructures.

References

1. Akella, M.R., Batta, R., Delmelle, E.M., Rogerson, P.A., Blatt, A., Wilson, G.: Base Station Location and Channel Allocation in a Cellular Network with Emergency Coverage Requirements. *European Journal of Operational Research* 164(2), 301–323 (2005)
2. Alba, E.: Evolutionary Algorithms for Optimal Placement of Antennae in Radio Network Design. In: International Parallel and Distributed Processing Symposium, vol. 7, p. 168 (2004)
3. Alba, E.: Parallel Metaheuristics: A New Class of Algorithms. Wiley-Interscience, Hoboken (2005)

4. Brockhoff, D., Friedrich, T., Hebbinghaus, N., Klein, C., Neumann, F., Zitzler, E.: Do Additional Objectives Make a Problem Harder? In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, GECCO 2007, pp. 765–772. ACM, New York (2007)
5. Coello, C.A., Lamont, G.B., Veldhuizen, D.A.V.: Evolutionary Algorithms for Solving Multi-Objective Problems. In: Genetic and Evolutionary Computation (2007)
6. Handl, J., Lovell, S.C., Knowles, J.: Multiobjectivezation by Decomposition of Scalar Cost Functions. In: Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N. (eds.) PPSN 2008. LNCS, vol. 5199, pp. 31–40. Springer, Heidelberg (2008)
7. Knowles, J.D., Watson, R.A., Corne, D.: Reducing Local Optima in Single-Objective Problems by Multi-objectivization. In: Zitzler, E., Deb, K., Thiele, L., Coello Coello, C.A., Corne, D.W. (eds.) EMO 2001. LNCS, vol. 1993, pp. 269–283. Springer, Heidelberg (2001)
8. Mendes, S.P., Molina, G., Vega-Rodríguez, M.A., Gómez-Pulido, J.A., Sáez, Y., Miranda, G., Segura, C., Alba, E., Isasi, P., León, C., Sánchez-Pérez, J.M.: Benchmarking a Wide Spectrum of Meta-Heuristic Techniques for the Radio Network Design Problem. *IEEE Trans. Evol. Comput.*, 1133–1150 (2009)
9. Mendes, S.P., Pulido, J.A.G., Rodriguez, M.A.V., Simon, M.D.J., Perez, J.M.S.: A Differential Evolution Based Algorithm to Optimize the Radio Network Design Problem. In: E-SCIENCE 2006: Proceedings of the Second IEEE International Conference on e-Science and Grid Computing, p. 119. IEEE Computer Society, Washington, DC, USA (2006)
10. Meunier, H., Talbi, E.G., Reininger, P.: A Multiobjective Genetic Algorithm for Radio Network Optimization. In: Proceedings of the 2000 Congress on Evolutionary Computation, pp. 317–324. IEEE Press, Los Alamitos (2000)
11. Segura, C., González, Y., Miranda, G., León, C.: A Multi-Objective Evolutionary Approach for the Antenna Positioning Problem. In: Setchi, R., Jordanov, I., Howlett, R., Jain, L. (eds.) KES 2010. LNCS, vol. 6276, pp. 51–60. Springer, Heidelberg (2010)
12. Segura, C., Segredo, E., González, Y., León, C.: Multiobjectivisation of the Antenna Positioning Problem. In: Abraham, A., Corchado, J., González, S., De Paz Santana, J. (eds.) International Symposium on Distributed Computing and Artificial Intelligence. AISC, vol. 91, pp. 319–327. Springer, Heidelberg (2011)
13. Talbi, E.G., Meunier, H.: Hierarchical Parallel Approach for GSM Mobile Network Design. *J. Parallel Distrib. Comput.* 66(2), 274–290 (2006)
14. Wan Tcha, D., Myung, Y.S., Hyuk Kwon, J.: Base Station Location in a Cellular CDMA System. *Telecommunication Systems* 14(1-4), 163–173 (2000)
15. Veldhuizen, D.A.V., Zydallis, J.B., Lamont, G.B.: Considerations in Engineering Parallel Multiobjective Evolutionary Algorithms. *IEEE Trans. Evol. Comput.* 7(2), 144–173 (2003)
16. Weicker, N., Szabo, G., Weicker, K., Widmayer, P.: Evolutionary Multiobjective Optimization for Base Station Transmitter Placement with Frequency Assignment. *IEEE Trans. Evol. Comput.* 7(2), 189–203 (2003)

Adaptive Request Distribution in Cluster-Based Web System

Krzysztof Zatwarnicki

Department of Electrical, Control and Computer Engineering
Opole University of Technology, Opole, Poland
k.zatwarnicki@gmail.com

Abstract. This work presents the application of two adaptive decision making algorithms in Web switch controlling operations of cluster-based Web systems. The proposed approach applies machine learning techniques; namely a fuzzy logic and neural networks, to deploy the adaptive and intelligent dispatching within locally distributed fully replicated Web servers. We present the adaptive control framework and the design of Web switch applying our conception. We demonstrate through the simulations experiments the difference between our algorithms and compare them with the most popular and reference distribution algorithms.

Keywords: Quality of Web service, Cluster-based Web system, HTTP request distribution.

1 Introduction

Quality of Web service (QoWS) is nowadays one of the key elements which have a significant impact on the profitability of conducted Internet enterprises. There are many different ways of improving QoWS [6]. The encountered solutions include the application of a more efficient server in the service, scheduling and admission control in the Web server [12], the application of a locally [9] and globally [4] distributed cluster-based Web system.

The most commonly used technique for improving QoWS is the application of a locally distributed cluster-based Web system where a group of Web servers represented by one virtual IP address work together to service HTTP requests. The key element controlling the cluster-based Web system is a Web switch, which receives HTTP requests and distributes them sharing the load among Web servers. The request distribution algorithm in the Web switch decides which of the Web servers should service the request. The efficiency of the service and the request response time significantly depends on the type of applied algorithm. The most commonly applied in industrial solutions simple request distribution algorithms are Round-Robin (RR) and its variation Weighted Round-Robin (WRR) [6]. Other more complex algorithms take into account the content of the requests. Reference algorithms belonging to this group are for example Locality Aware Request Distribution (LARD) [1] taking into account past decisions and Content Aware Policy (CAP) [7] working like the RR algorithm

for each of the kind of requests. A separate group of distribution algorithms are adaptive algorithms which can adapt to a changing environment. The most well-known are AdaptLoad [13] which distribute the request on the base of the size of requested objects, Adaptive Load Balancing Mechanism (ALBM) [8], which take into account the load of the server applications and finally two algorithms, proposed by the author, minimizing request response time, Fuzzy Adaptive Request Distribution (FARD) [3] and Local Fuzzy Neural Requests Distribution (LFNRD). A good survey of different distribution algorithms can be found in [9].

In this paper two request distribution algorithms FARD and LFNRD are compared. The LFNRD is a new algorithm created on the base of the FARD algorithm. Each of the algorithms use in its construction fuzzy-neural approach. By applying the computational and learning potentials of neural networks into fuzzy systems we can achieve learning and adaptation characteristics of such hybrid solutions. The integration of neural networks and fuzzy logic leads to controllers with an adaptive nature that supports robust control in uncertain and noisy environments [4].

The rest of the paper is organized as follows. First, we introduce the idea of adaptive request distribution. Then, we show how the concepts of a fuzzy logic and neural networks are merged in two request distribution algorithms. After that, the simulation model and results are discussed. Finally, we present the conclusion.

2 Adaptive Request Distribution

An adaptive control system can be defined as a feedback control system intelligent enough to adjust its characteristics in a changing environment so as to operate in an optimal manner to some specific criteria [5]. Basically in an adaptive control scheme, we can assume that there exist a basic control algorithm and an adaptation algorithm used to gradually improve the basic algorithm.

In our approach we are controlling the operations of a locally distributed Web system. The following elements can be distinguished in the Web system: clients sending HTTP requests, Web switch redirecting incoming requests and Web servers servicing requests with the use of backend servers which include application and database servers. Fig. 1.a presents cluster-based Web system.

The purpose of the operation of the system in our conception is to minimize HTTP request response times in the way that for each individual HTTP request Web server offering the shortest response time is chosen. All the decisions and supervision under the system is realized by the Web switch.

We assume that the content offered in the service is fully replicated and each of the Web servers working in the cluster can service each request belonging to the set of requests accepted in the service. We also assume the requests in the web switch are serviced according to the First Come First Served policy and all HTTP requests are treated equally.

In order to describe the conception let us introduce the following denotations: x_i – HTTP request, $x_i \in X$, X is a set of requests serviced in the Web service; i – index of the HTTP request, $i = 1, \dots, I$; O_i – load of the system at the moment of i th

request arrival, $O_i = [O_i^1, \dots, O_i^s, \dots, O_i^S]$, $s = 1, \dots, S$; S – number of Web servers in the system; O_i^s – load of the s th Web server; \tilde{t}_i – response time for the i th request, measured from the moment of sending the request from the Web switch to the server, up to receiving the HTTP response by the switch; \hat{t}_i^s – estimated response time of the i th request for s th Web server, $s = 1, \dots, S$; w_i – decision, Web server chosen to service the i th request, $w_i \in \{1, \dots, S\}$; U_i^s – parameters of the s th Web server in the i th moment in decision making algorithm.

The concept of adaptive decision making used is illustrated in Figure 2. The decision making algorithm consists of two parts: the basic distribution algorithm and the adaptation algorithm. The decision making algorithm designate the decision $w_i = \min_s \{\hat{t}_i^s : s = 1, \dots, S\}$ taking into account the load O_i to estimate request response times $\hat{t}_i^1, \dots, \hat{t}_i^s, \dots, \hat{t}_i^S$. Based on the estimated response time $\hat{t}_i^s|_{s=w_i}$ and observed response time \tilde{t}_i , the adaptation algorithm calculates new parameters $U_{(i+1)}^s|_{s=w_i}$ of the basic distribution algorithm which are used in the next decision ($i + 1$).

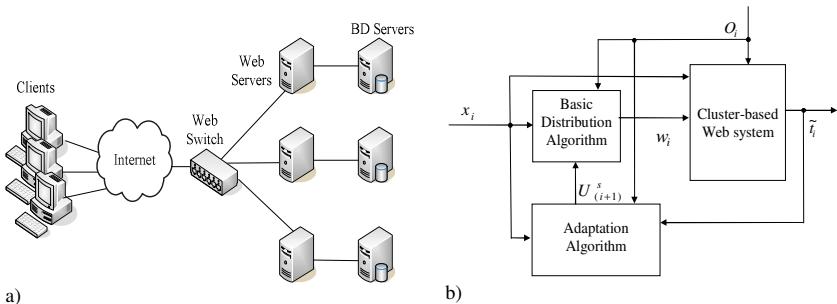


Fig. 1. a) Cluster-based Web system, b) Decision making process

Based on the presented conception we have developed two distribution algorithms namely LFNRD and FARD. Both of the algorithms use the same basic distribution algorithm and different adaptation algorithms. The proposed algorithms are built around the merged application of two intelligent decision making methods, namely fuzzy and neural networks. Design of Web switch applying discussed conception is presented in the next section.

3 Web Switch Design

The Web switch constructed in accordance with the concept of adaptive request distribution consists of three main parts: a distribution algorithm determining Web

server to service the request, an execution module which executes the decisions and a measurement module collecting informations required in decision-making process. The schema of adaptive Web switch is presented on the Fig. 2.

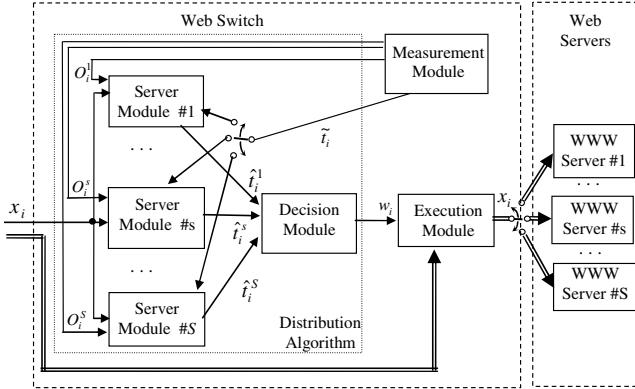


Fig. 2. Adaptive Web switch

The HTTP request is treated in the Web switch on one hand as the information referred to the task the Web server should carry out (marked as a single line on the figures) but on the other hand as the carrier of the information transferred physically to the Web server (marked as double line).

The incoming request x_i is at first redirected to the server models which estimates request response times \hat{t}_i^s , $s = 1, \dots, S$, for each Web server operating in the cluster. Each server model represents one Web server. The time \hat{t}_i^s is estimated on the base of the load O_i^s of the Web server. The description of the operation of the server model will be presented in a later part of this section.

The decision module determines the decision w_i according to $w_i = \min_s \{\hat{t}_i^s : s = 1, \dots, S\}$, where $w_i \in \{1, \dots, S\}$. The Web server offering the shortest estimated request response time is chosen.

The execution module supervises the servicing of the request on the Web server. It transfers the request x_i to the chosen Web server w_i , receives the response and transfers the response to the client.

During servicing the request by the Web server the measurement module measures the response time \tilde{t}_i , and after the completion of the service passes the time to the w_i th server module. The measurement module determines also the load of each of the Web servers O_i^s , $s = 1, \dots, S$, where $O_i^s = [a_i^s, b_i^s]^T$, a_i^s is the number of static requests and b_i^s is the number of dynamic requests concurrently serviced by the s th

Web server. All the information collected by the module is available within the Web switch.

3.1 Server Model

The way of operation of the server module is different for the FARD and the LFNRD algorithm. However, the overall structure of functional blocks of the server module for both algorithms is similar. The following modules can be distinguished in the server model: classification, estimation mechanism, adaptation mechanism and parameter DB (Fig. 3.a).

For clarity of denotations the index s of the server will be dropped in the pertaining formulas.

The classification module classifies all incoming requests. The class k_i of the requested object is determined on the base of the object's size, in the case of static objects where every dynamic object has its own individual class, $k_i \in \{1, \dots, K\}$ and K is the number of classes. Objects belonging to the same class have similar service times.

The parameter DB module stores parameters of the server model. It was marked as U_i for the LFNRD algorithm and U'_i for the FARD algorithm. There are different sets of parameters for objects belonging to different classes therefore $U_i = [U_{1i}, \dots, U_{ki}, \dots, U_{Ki}]^T$ and $U'_i = [U'_{1i}, \dots, U'_{ki}, \dots, U'_{Ki}]^T$, $k \in \{1, \dots, K\}$.

The estimation mechanism estimates response time \hat{t}_i for request x_i based on the load O_i^s of the server taking in to account the parameters U_{ki} or U'_{ki} . The adaptation mechanism updates parameters $U_{k(i+1)}$ or $U'_{k(i+1)}$ on the base of the load O_i^s , estimated \hat{t}_i and measured \hat{t}_i response times, and previous values of parameters U_{ki} or U'_{ki} .

The estimation and adaptation modules for both the FARD and the LFNRD algorithms form neuro-fuzzy models in which the following blocks can be distinguished: fuzzification, inference and defuzzification. Both of the models are different, however they estimate the request response time in the same way and adapt to the time-varying environment in a different way. The Fig 3.b presents the neuro-fuzzy model for the FARD algorithm while Fig 3.e presents the model for the LFNRD algorithm.

The inputs for both of the fuzzy-neural networks is the load of the server a_i and b_i . The fuzzy sets for input a_i are denoted as $Z_{a1}, \dots, Z_{al}, \dots, Z_{aL}$, and similarly fuzzy sets for input b_i are denoted as $Z_{b1}, \dots, Z_{bm}, \dots, Z_{bM}$, where L and M are the number of fuzzy sets. The fuzzy sets membership functions for both inputs are denoted as $\mu_{Z_{al}}(a_i)$, $\mu_{Z_{bm}}(b_i)$, $l = 1, \dots, L$, $m = 1, \dots, M$. The functions are triangular and can be presented like on the Fig. 3.c. The parameters $\alpha_{1ki}, \dots, \alpha_{lki}, \dots, \alpha_{(L-1)ki}$ specify the shape of membership functions for input a_i and

$\beta_{1ki}, \dots, \beta_{mki}, \dots, \beta_{(M-1)ki}$ for input b_i . The outputs of fuzzification blocks are the values of membership to individual fuzzy sets.

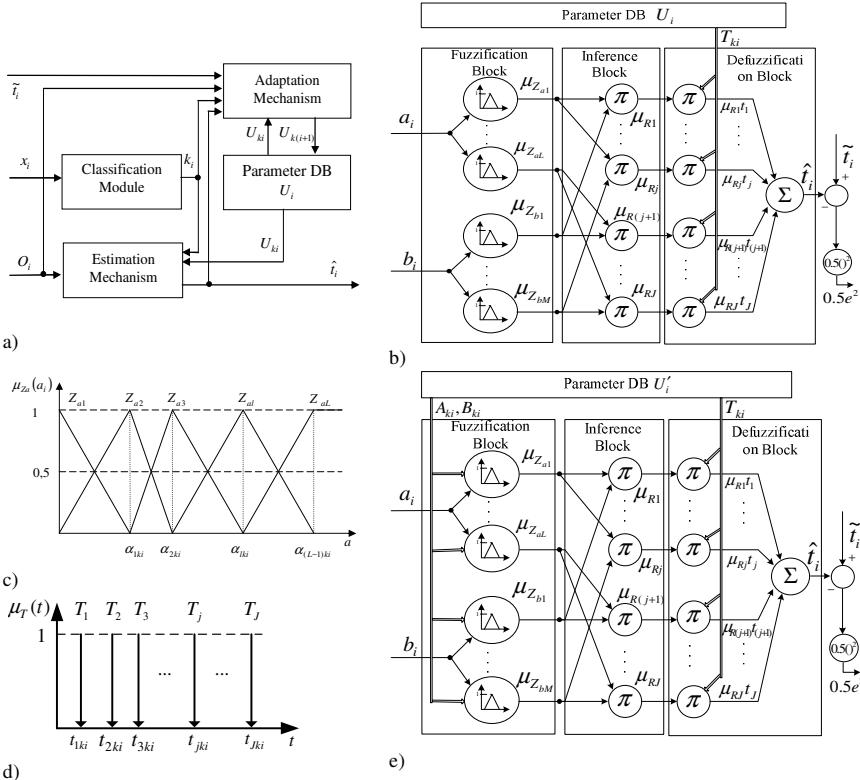


Fig. 3. a) Server model module, b) Neuro-fuzzy model for FARD algorithm, c) Membership functions for input, d) Membership functions for output, e) Neuro-fuzzy model for LFNRD algorithm

In the inference block products of inference are obtained $\mu_{Rj} = \mu_{Z_{al}}(a_i) \cdot \mu_{Z_{bm}}(b_i)$, where $l = 1, \dots, L, m = 1, \dots, M, j = 1, \dots, J, J = M \cdot L$. The rule base used in the module consist of J rules constructed in following way: $R_j : \text{IF } (a = Z_{al}) \text{ AND } (b = Z_{bm}) \text{ THEN } (t = T_j)$, where a, b denotes load of the server, t is a response time, $T_1, \dots, T_j, \dots, T_J$ are fuzzy sets for output \hat{t}_i , $l = 1, \dots, L, m = 1, \dots, M, j = 1, \dots, J$.

In the defuzzification block the estimated service time value \hat{t}_i is calculated with use of the Height Method. Assuming that membership functions for outputs fuzzy sets are singletons indicating values $t_{1ki}, \dots, t_{jki}, \dots, t_{Jki}$ (Fig. 3.d) the estimated service time can be calculated in the following way $\hat{t}_i = \sum_{j=1}^J t_{jki} \mu_{R_j}$.

In the adaptation process the FARD algorithm modifies only the parameters of the output membership functions $T_{ki} = [t_{1ki}, \dots, t_{jki}, \dots, t_{Jki}]$ therefore the parameters DB can be presented as follow $U_{ki} = T_{ki}$. In this algorithm the parameters for input fuzzy sets membership functions $A_{ki} = [\alpha_{1ki}, \dots, \alpha_{lki}, \dots, \alpha_{(L-1)ki}]$, $B_{ki} = [\beta_{1ki}, \dots, \beta_{mki}, \dots, \beta_{(M-1)ki}]$ are constant and uniformly cover the span of the inputs a_i and b_i . In the LFNRD algorithm not only the parameters of output fuzzy sets are modified $T_{ki} = [t_{1ki}, \dots, t_{jki}, \dots, t_{Jki}]$ but also parameters of input fuzzy sets $A_{ki} = [\alpha_{1ki}, \dots, \alpha_{lki}, \dots, \alpha_{(L-1)ki}]$, $B_{ki} = [\beta_{1ki}, \dots, \beta_{mki}, \dots, \beta_{(M-1)ki}]$, therefore $U'_{ki} = [A_{ki}, B_{ki}, Y_{ki}]$.

The backpropagation method introduced by Werbos [11] and gradient descent rule developed by Widrow and Hoff were used to tune the parameters. According to this, the parameters are adjusted based on the general formula $\delta_{(i+1)} = \delta_i - \eta \partial E_i / \partial \delta_i$ where $\delta_{(i+1)}$ is the updated value of the parameter δ_i , η is the learning rate, and $\partial E_i / \partial \delta_i$ is the partial error, $E_i = (e_i)^2 / 2$ and $e_i = \hat{t}_i - \tilde{t}_i$. Based on this new values of parameters for the output membership functions are calculated in the FARD and the LFNRD algorithms according to $t_{jki(i+1)} = t_{jki} + \eta_t \mu_{R_j} (\tilde{t}_i - \hat{t}_i)$, where η_t is adaptation ratio.

The parameters of input fuzzy sets membership functions in the LFNRD algorithm are updated in the following way:

$$\alpha_{lki(i+1)} = \alpha_{lki} + \eta_a (\tilde{t}_i - \hat{t}_i) \sum_{\gamma=1}^M \left(\mu_{Z_{by}}(b_i) \sum_{\varphi=1}^L \left(t_{((m-1)L+\varphi)ki} \partial \mu_{Z_{a\varphi}}(a_i) / \partial \alpha_{lki} \right) \right),$$

$$\beta_{mki(i+1)} = \beta_{mki} + \eta_b (\tilde{t}_i - \hat{t}_i) \sum_{\varphi=1}^L \left(\mu_{Z_{a\varphi}}(a_i) \sum_{\gamma=1}^M \left(t_{((l-1)M+\gamma)ki} \partial \mu_{Z_{by}}(b_i) / \partial \beta_{mki} \right) \right),$$

where η_a, η_b are adaptation ratios, $l = 1, \dots, L-1$, $m = 1, \dots, M-1$.

4 Simulation Model and Experiment Results

In order to evaluate the operations of Web cluster operating under control the FARD and the LFNRD algorithms simulation experiments were conducted with the use of the CSIM 19 package [10]. The simulation program consists of the following modules: HTTP request generator, Web switch, Web and database servers (Fig. 4a).

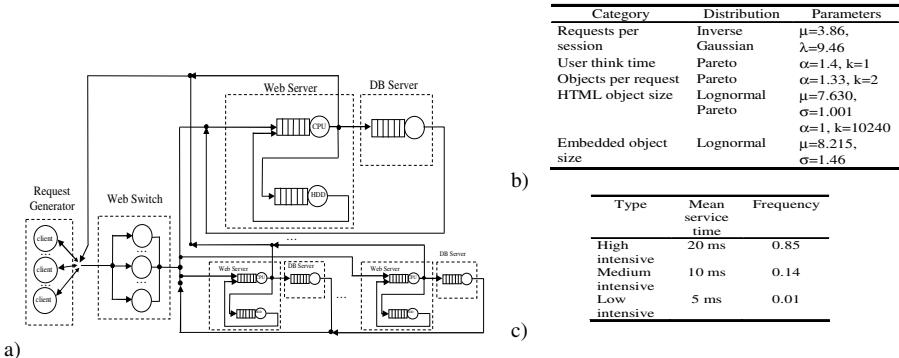


Fig. 4. a) Simulation model, b) Workload model parameters, c) Workload model parameters of dynamic objects

In the Web switch module the following algorithms were implemented FARD and LFNRD, the most popular RR and WRR, reference most effective algorithms LARD and CAP. The server model consisted of the processor, the hard drive and the cache memory. The service times were established on the base of experiments for the Apache Web server operating on a computer with Intel Pentium 4, a 2 GHz processor and a Seagate ST340810A IDE hard drive. The service of static and dynamic requests was provided. The dynamic requests, serviced by the Web and database server, were divided into three classes [7]: high intensive, medium intensive and low intensive. The service times of the dynamic requests were modeled according to hyperexponential distribution with parameters presented in Fig. 4c. The request generator module was working in the way that the generated request traffic complied with the traffic observed on the Internet, which is characterized by bursts and self-similarity [2] (Fig. 4b).

The experiments were conducted for two Web clusters, the first one contained three set of Web and database servers, the second one contained five sets of servers. In the Fig. 5.a and 5.b the diagrams of mean request response time in function of the load (new clients per second) are presented.

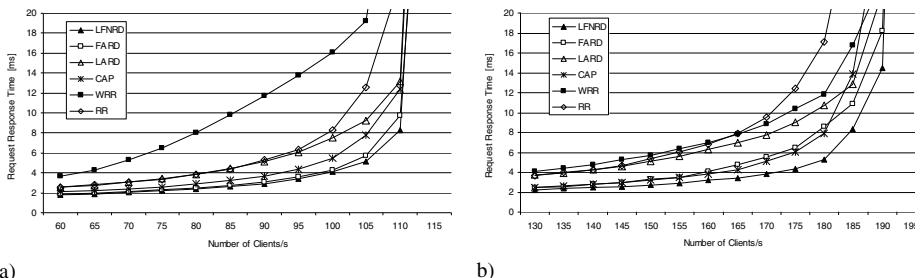


Fig. 5. Request response time vs. load a) three Web servers, b) five Web servers

As one can see the shortest request response times were obtained for the LFNRD algorithm both for three and five Web server clusters. Slightly longer times especially

for five Web server cluster were obtained for the FARD algorithm. The differences in the working of the algorithms are connected with an adopted method of adaptation to a time-varying environment. On the Fig. 6 diagrams are presented of relative error for FARD and LENRD algorithms in the function of the number of serviced requests for all kinds of objects (Fig. 6.a, 6.b) and big objects (>50KB) (Fig. 6.c, 6.d).

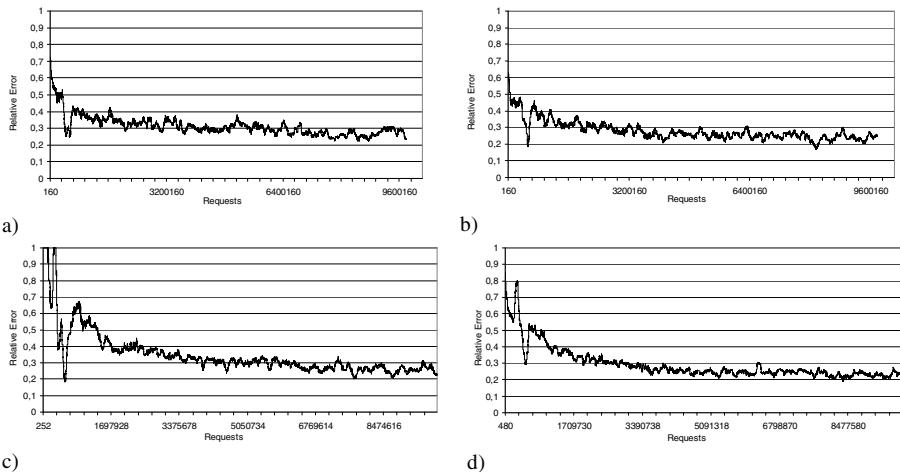


Fig. 6. Relative error vs. number of serviced requests a) all requests, FARD; b) all requests, LFNRD; c) big static objects, FARD; d) big static objects, LFNRD

In the process of adaptation, the LFNRD algorithm adapted better to the way of operation of the servers in the cluster than the FARD algorithm. After servicing about 5 Mio requests the LFNRD algorithm is fully trained and imitates well operations of servers. The FADR algorithm requires servicing about 7 Mio request to be trained. The mean relative error for all kinds of requests for fully trained LFNRD algorithm is 0.27, the standard deviation is 0.19 and the median 0.23 while for FARD algorithm the values are: mean 0.3, standard deviation 0.22, median 0.26. Both algorithms differently estimated request response times for different types of objects. For LFNRD algorithm the mean, standard deviation and median values for small static objects (<50KB) were 0.26, 0.19, 0.23, for big objects (>50KB) were 0.26, 0.19, 0.23 and for dynamic objects were 0.31, 0.21, 0.27. For the FARD algorithm and small objects the following values were obtained 0.3, 0.22, 0.26, for big objects 0.3, 0.22, 0.25 and for dynamic objects 0.47, 0.29, 0.45. As it can be noticed the biggest values of relative error were obtained for the FARD algorithm especially for dynamic objects, therefore, the quality of service in the case of the FARD algorithm is lower than for the LFNRD algorithm. Further experiments have however shown, that the employment of labour consuming the distribution algorithm can significantly affect the mean decision taking time. In the experiments the server with two Intel Dual-Core Xeon 5160 processors and the software with a no optimized code of the FARD and the LFNRD algorithms were used. The mean decision time for the FARD algorithm was 0.22 μ s and for the LFNRD algorithm was 1.6 μ s while the number of decisions taken per second for the

FARD was 4,5 Mio and for the LRNFD 616,000. The results have shown that both algorithms can operate in real time even in a heavily loaded Web system.

5 Summary

In the presented paper the problem of request distribution with the use of adaptive decision making algorithms was discussed. Two request distribution algorithms namely FARD and LFNRD minimizing HTTP request response times were presented. Both of the algorithms apply in its' construction neuro-fuzzy models to estimate requests response times offered by Web servers operating in the cluster. Experiments have shown that the application of a more complex neuro-fuzzy model can increase the quality of Web service, however the decision making time is much longer for complex algorithm.

References

1. Aron, M., Sanders, D., Druschel, P., Zwaenepoel, W.: Scalable content-aware request distribution in cluster-based network servers. In: Proc. of USENIX 2000 Conf., USA (2000)
2. Barford, P., Crovella, M.E.: A Performance Evaluation of Hyper Text Transfer Protocols. In: Proc. ACM SIGMETRICS 1999, Atlanta, pp. 188–197 (1999)
3. Borzemski, L., Zatwarnicki, K.: A Fuzzy Adaptive Request Distribution algorithm for cluster-based Web systems. In: 11th Euromicro Workshop on Parallel, Distributed and Network-Based Processing (PDP 2003), pp. 119–126 (2003)
4. Borzemski, L., Zatwarnicki, K., Zatwarnicka, A.: Adaptive and Intelligent Request Distribution for Content Delivery Networks. *Cybernetics and Systems* 38(8), 837–857 (2007)
5. Bubnicki, Z.: Modern Control Theory. Springer, Berlin (2005)
6. Cardellini, V., Casalicchio, E., Colajanni, M., Yu, P.S.: The state of the art in locally distributed Web-server systems. *ACM Computing Surveys* 34(2), 263–311 (2002)
7. Casalicchio, E., Colajanni, M.: A Client-Aware Dispatching Algorithm for Web Clusters Providing Multiple Services. In: Proc. of the 10th International World Wide Web Conference, Hong Kong (2001)
8. Choi, E.: Performance test and analysis for an adaptive load balancing mechanism on distributed server cluster systems. *Future Generation Systems* 20, 237–247 (2004)
9. Gilly, K., Juiz, C., Puigjaner, R.: An up-to-date survey in web load balancing. Springer, Heidelberg (2010), World Wide Web, 10.1007/s11280-010-0101-5
10. Mesquite Software Inc. CSIM User's Guide, Austin, TX (2010),
<http://www.mesquite.com>
11. Werbos, P.: Beyond regression: New tools for prediction and analysis in the behavioral sciences, Ph.D. dissertation, Committee on Appl. Math., Cambridge (1974)
12. Zatwarnicki, K.: Providing Web Service of Established Quality with the Use of HTTP Requests Scheduling Methods. In: Jędrzejowicz, P., Nguyen, N.T., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2010. LNCS (LNAI), vol. 6070, pp. 142–151. Springer, Heidelberg (2010)
13. Zhang, Q., Riska, A., Sun, W., Smirni, E., Ciardo, G.: Workload-aware load balancing for clustered web servers (2005)

Globally Evolved Dynamic Bee Colony Optimization

Anggi Putri Pertiwi and Suyanto

The faculty of Informatics - Telkom Institute of Technology,
Jl. Telekomunikasi No. 1 Terusan Buah Batu, Bandung 40257, West Java, Indonesia
anggi.putripertiwi@yahoo.com, suy@ittelkom.ac.id

Abstract. Bee colony optimization (BCO) is one of swarm intelligence algorithms that evolve static and locally. It performs slow improvement and tends to reach a local solution. In this paper, three modifications for the BCO are proposed, i.e. global evolution for some bees, dynamic parameters of the colony, and special treatment for the best bee. Computer simulation shows that Modified BCO performs quite better than the BCO for some job shop scheduling problems.

Keywords: optimization, bee colony, global evolution, dynamic parameters, job shop scheduling.

1 Introduction

Bee Colony Optimization is inspired by a natural phenomenon, i.e. foraging behavior of honey bee colony. In BCO proposed by Teodorović [1], [5], [6], some bees in a population search and explore the sources having the most food. After that, the bees will return to the hive and share the information about the amount of the food and the proximity of the food sources to the hive. The way of sharing this information is by performing a kind of dancing called “waggle dance”. The sources that have more foods will be exploited by more bees. The BCO has been successfully used for solving some combinatorial optimization, i.e. Travelling Salesman Problem (TSP), Ride-Matching Problem (RMP), Routing and Wavelength Assignment (RWA) in All-Optical Networks, the p -median problem, static scheduling of independent tasks on homogenous multiprocessor systems, and traffic sensors location problem on highways [1], [5], [6].

The BCO has three main characteristic. Firstly, it locally evolves in each generation since there is no solution in a generation that is exploited to make an improved solution in the next generation. Secondly, BCO uses static parameters in each generation. Thirdly, BCO only adopts the foraging behavior of honey bee colony to get a solution. These characteristics make BCO performs slow improvement and can be trapped at a local minimum.

In this paper, three modifications for the BCO are proposed to improve its performance. Firstly, the modified BCO (MBCO) is designed to globally evolve using some first partial solutions in a generation to be exploited in the next generation. Secondly, MBCO uses dynamic parameters, i.e. number of constructive moves and probability

to keep solution, in each generation. Thirdly, a special improvement move, using Tabu Search, is performed to a bee producing the best solution.

Some instances of Job Shop Scheduling Problem (JSP) will be used to compare the performances of BCO and MBCO. JSP concerns on resources allocation to fulfill the customer's demand by optimizing a particular objective function [2]. This problem is common in manufacture industries. The schedule optimizing the resources utilization will increase production and decrease the cost. JSP has some constraints, especially the precedence constraints where the operations on a particular job must be processed in sequence. The goal is to find a schedule that satisfies the constraints and give minimum completion time (*makespan*) [2], [4].

2 Bee Colony Optimization

In nature, honeybee colony has some intelligent behaviors, e.g. foraging behavior, marriage behavior, queen bee concept. The BCO from [1], [5], [6] adopts foraging behavior in bee colony. To find a solution, BCO has two stages, forward pass and backward pass. In the forward pass, each bee in the colony partially generates a solution. In the backward pass, the bees evaluate and change the information about quality of the solution they found. To find a better solution, a bee will focus on more promising areas and abandon solutions on less promising ones. Both forward pass and backward pass are alternately performed until meet stopping criteria [5].

2.1 Local Evolution

In the original BCO proposed by Teodorovic [1], [5], [6], there is no procedure to exploit the complete solutions in a generation to be improved in the next generation. After constructing complete solutions in a cycle of the local evolution, using forward pass and backward pass, the bees preserve the best-so-far solution and forget the others. Then, the bees start to generate new complete solutions. This process is performed repeatedly. Hence, the evolution is locally performed in one generation only. In forward pass, any bee independently constructs a partial solution. But, in backward pass, a bee producing worse solution will follow another bee having better solution. So, a “generation” in BCO is more precisely called a “trial” since all bees just repeat the same processes.

2.2 Static Parameters

The BCO has two parameters, i.e. number of bees (B) and number constructive moves (NC). Both parameters are static during the evolution. So, the bees try to find solutions using the same strategy in whole generations. This strategy produces a stagnant evolution in some generation. Hence, BCO tends to reach a local optimum.

In BCO, the probability of b -th bee to keep its solution is calculated by the formula

$$Pb^{u+1} = e^{-\left(\frac{O_{max}-O_b}{u}\right)} \quad b = 1, 2, \dots, B, \quad (1)$$

where u = the ordinary number of the forward pass (e.g., $u = 1$ for the first forward pass, $u = 2$ for the second forward pass, and so on), O_b is a normalized objective

function value of the b -th bee, and O_{max} is the maximum normalized objective value among all bees. In this case, the objective function is calculated from the total processing time of all operations subtracted by *makespan* of the schedule. Hence, the objective value is positive since the *makespan* is less than the total processing time. The probability formula in Equation (1) produces the percentage of number of bees to keep their solution, in each generation, will be relatively static (see Figure 1).

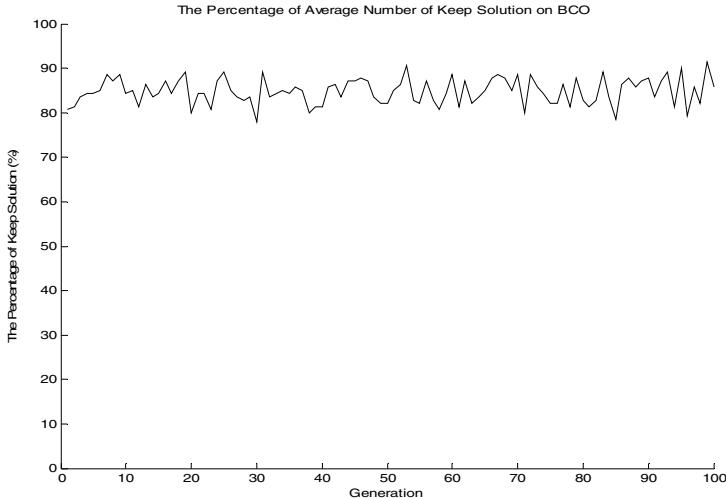


Fig. 1. The percentage of average number of bee to keep its solution during a local evolution, forward pass and backward pass, performed by BCO

3 Three Modifications for Bee Colony Optimization

Three modifications are proposed to overcome the BCO's disadvantages. The first modification is to use some first partial solutions generated by some bees to be exploited in the next generation so that the MBCO evolve globally. The second one is to use dynamic parameters in each generation. In the original BCO, starting the process to search the solution from the beginning by using static parameters for each generation is like repeating the process with the same strategies. By using dynamic parameters, MBCO tries to do different process for each generation. Furthermore, the parameters can be designed to evolve globally. The third one is to combine the foraging behavior in BCO with other bee's intelligent behavior. In nature, a queen is the biggest bee in the colony. It is developed from a selected larva that gets more and better foods from the worker bees than the others. So, the queen candidate will grow to be the best bee in the colony. That behavior can be adopted to improve the performance of BCO. It can be assumed that a bee in nature finding the best foods for the queen candidate will get a special treatment. In MBCO, an artificial bee producing the best solution will be improved by a local search. The improved solution is expected to be

the best solution and its partial solution can be used in the next generation. With these modifications, MBCO improves solution more quickly. The pseudo code of MBCO is described below:

-
1. Initialization
 - a. Generate a solution for a bee using shortest processing time method (SPT)
 - b. Generate $B-1$ solutions using MBCO for one generation
 - c. Population = B bees, one solution from TS and $B-1$ solutions from MBCO
 - d. Find n best solutions to be exploited in the next generation
 2. Improve the best solution from the previous generation using TS
 3. For each $B-1$ bee: //the forward pass
 - i. Set $k = 1$; //counter for constructive moves in the forward pass;
 - ii. Evaluate all possible constructive moves;
 - iii. According to evaluation, choose one move using random procedure;
 - iv. $k = k + 1$; If $k \leq NC$ **Goto** step ii.
 4. All bees are back to the hive; //backward pass starts;
 5. If $B-1$ solutions are not completed **Goto** step 3
 6. Join one solution from TS to the $B-1$ solutions
 7. Evaluate B solutions and find n best solutions to be exploited in the next generation
 8. If the stopping condition is not met **Goto** step 2
 9. Output the best solution found
-

Figure 2 illustrates the first generation of the MBCO. Initially, *Bee 1* as the special bee is separately from the colony and gets the special treatment to be modified by TS. The colony that consists of *Bee 2*, *Bee 3*, and *Bee 4* construct the partial solution in the first forward pass. After that, the bees back to the hive and do the second stage, called backward pass. In the first backward pass, *Bee 3* and *Bee 4* decide to keep their first partial solutions, while *Bee 2* decides to abandon its already generated partial solution and to follow the first partial solution of *Bee 3*. After copying *Bee 3*'s first partial solution, *Bee 2* is free to choose the next constructive move. These two stages are alternating in order to generate the complete solution. Through the backward pass, a bee with bad solution can improve its solution by follow a better solution generated by another bee. This shows that the evolution is locally occurred in one generation. At the end of generation, *Bee 1* joins the colony and then be selected as the best bee. The best bee will be modified by TS in the second generation. The first partial solution, with length of 1 NC , produced by *Bee 2* and *Bee 4* will be used for the second generation. It shows the global evolution, where the predefined number of good previous first partial solutions are exploited in the next generation.

Figure 3 illustrates the second generation of the MBCO. *Bee 1* as the best bee from the first generation is modified by TS. Initially, both *Bee 2* and *Bee 3* use the two of 1 NC partial solutions from the first generation to generate partial solution in the first forward pass. In each backward pass, the bees do the recruiting process. As shown in Figure 3, in the last backward pass, both *Bee 3* and *Bee 4* keep their second partial solutions, while *Bee 2* decides to follow the second partial solution of *Bee 4*. At the end of generation, *Bee 1* joins the colony and selected as the best bee again. The first partial solution, with length of 1 NC , produced by *Bee 1* and *Bee 2* will be exploited in the next generation.

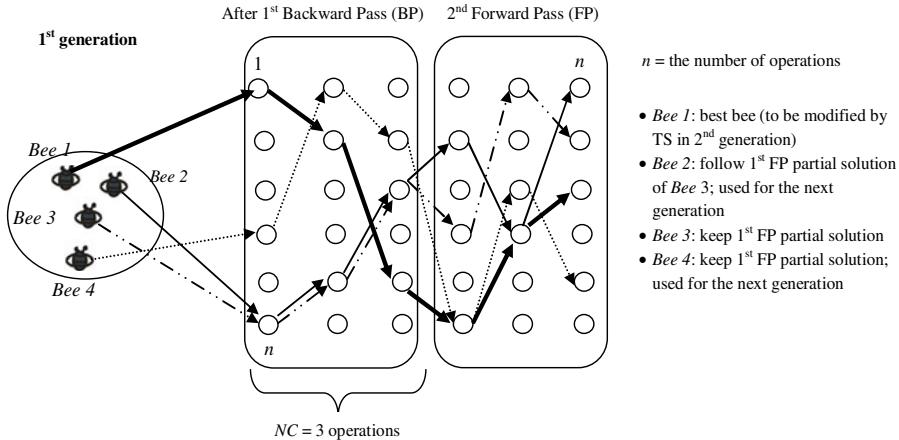


Fig. 2. The first generation of the local evolution in MBCO with 2 forward passes and $NC = 3$ operations

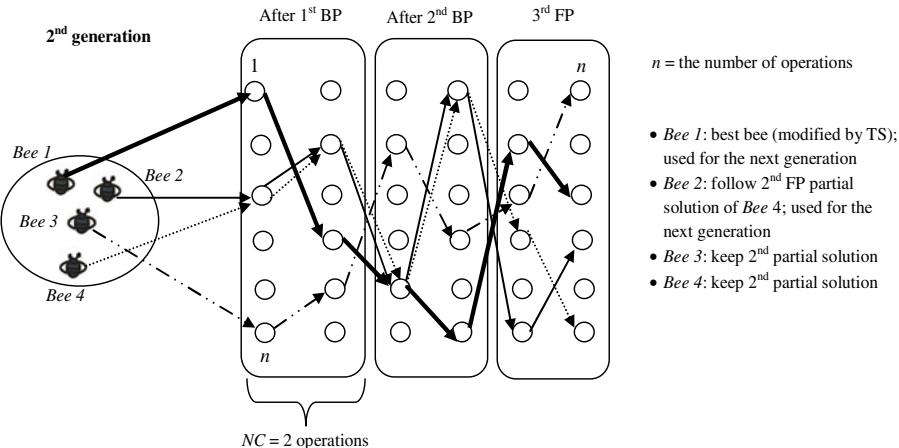


Fig. 3. The second generation of the evolution in MBCO with 3 forward pass and $NC = 2$ operations

3.1 Global Evolution

MBCO evolves globally since it uses some previous good solutions to be exploited in the next generation. The number of the previous solutions to be exploited is defined by a particular percentage of all solutions in the population. Some bees will use the first partial solutions of some good complete solutions from the previous generation which the length can be determined by a certain parameter. In the next generation, the bees will start the forward pass from the end of those partial solutions and try to find the new solutions.

A good complete solution with the minimum *makespan* can be obtained from a good partial solution too. It is possible that some good partial solutions will be potential to be reused. The bees are expected to find the complete solution as good as or even better from the previous one.

3.2 Dynamic Parameters

Different from BCO that uses static parameters, MBCO exploits two parameters that are dynamic during the evolution, i.e. NC and the probability of keep solution. In each generation, MBCO uses different NC , which is randomly generated in the predefined interval $[NC_{min}, NC_{max}]$. This allows the bee colony to find solutions using different strategy in each generation. Of course it will increase the solution diversity and expected to reach a better solution more quickly.

In Equation (1), the variable u is used to vary the probability. The probability of each bee to keep its solution increases when u increases. Hence, the number of bees that keep their solution increases when it comes close to the final forward pass. This tends to produce high diversity solutions. It does not focus on potential area that closes to optimum solution. Therefore, MBCO slightly modify the probability function into

$$Pb^{u+1} = e^{-\left(\frac{c*(O_{max}-O_b)}{U_{max}-u}\right)}, \quad (2)$$

where c is a control variable to adjust the probability. The probability decreases when c increases (see Fig 4). U_{max} is maximum number of forward pass in one generation. The other variables are the same as in the Equation (1).

In the Equation (2), c is designed to be dynamical so that allows the bees to evolve dynamically, generation by generation. In some first generations, any bee tends to keep its solution so that the diversity of partial solution is very high. But, in the last generation, the bees focus on exploiting some potential solutions. This is illustrated by Figure 5.

3.3 Special Treatment for the Best Bee

Another intelligent behavior of bee is queen bee concept. A queen is developed from larvae selected by worker bees and exclusively fed by high quality food called royal jelly, in order to become sexually mature. This behavior is adopted by MBCO to give a special treatment for a artificial bee producing best solution. The best bee, selected from the colony, is treated by improving its solution with local search technique. Hence, the improvement resulted by the best bee will be a better solution. In each generation, one best solution always selected to be improved by a local search.

In this paper, TS is used as local search technique. TS is a search procedure that capable of solving problems of combinatorial optimization by searching neighborhood of given solution [3]. TS limits the searching by prohibiting the bad solution which called “tabu” in a tabu list in order to avoid being trapped at a local minimum. The TS used here is the same as in [3].

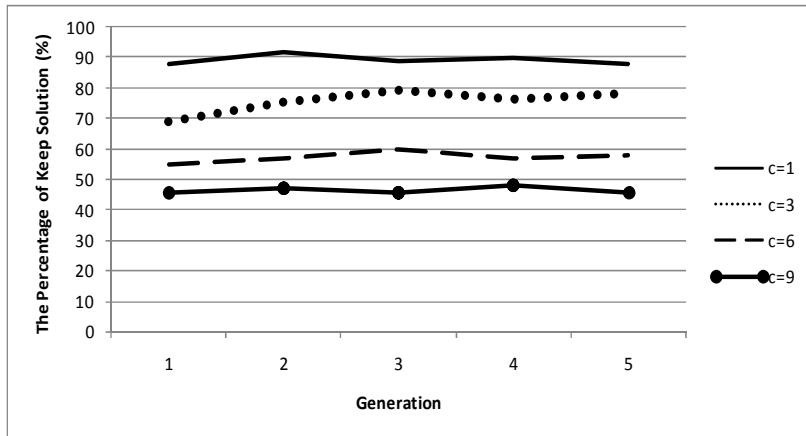


Fig. 4. The percentage of average number of bees to keep its solution during local evolution using the probability function in Equation (2)

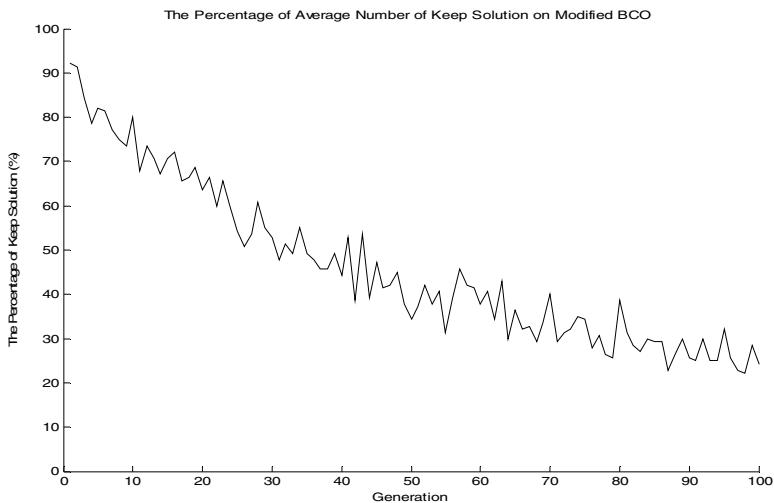


Fig. 5. The Percentage of average number of bee to keep its solution performed by MBCO using $c = 1$ to $c = 20$ and increment of 0.19 (calculated by $(20 - 1) / 100$)

4 Experiments and Results

In this study, MBCO is applied in the job shop scheduling problem (JSP). Computer simulation is focused on 10 JSP instances in order to compare its performance to the original BCO. The 10 JSP instance used are [2]:

- 2 problems from Fisher & Thompson (1963), ft06 and ft20;
- 2 problems from Adams (1988), abz5 and abz8; and
- 6 problems from Lawrence (1984), la03, la09, la21, la29, la31, and la37.

4.1 Job Shop Scheduling Problem

JSP contains a set of J from n jobs should be processed on a set of M from m machine. Any job J_i should be processed on any machine and consists of sequences of m_i operation, i.e. $O_{i1}, O_{i2}, \dots, O_{im_i}$, which have been scheduled before starting production process. O_{ij} is j -th operation of job J_i processed on a machine M_x in a processing time period τ_{ij} without interruption and preemption. A machine can only process a job, and a job can be processed by a machine in a time. The longest duration in all operation from all jobs is called *makespan* [2], [4].

A solution representation of JSP used is *preference list-based* [4]. A critical path, which is a sequence of operation that develops *makespan*, can be defined from a schedule [3]. It is very important to define neighborhood structure used to modify a solution in TS algorithm. Once the critical path is found, TS can modify a solution by inserting a move on between two ordered operations in the critical path, but keeping the precedence constraint.

4.2 Performance Analysis

In order to see the effectiveness of the three proposed modifications, both BCO and MBCO are applied to solve a JSP instance, i.e. la09. Figure 6 shows their dynamics of evolution for 250 generations. Generally, the figure shows that MBCO is more dynamic than BCO. MBCO generally makes improvement for the solution in each generation, although gets stuck for some generations, and finally reach a global optimum. But, BCO frequently gets stuck and reach a local minimum.

In the early generations, MBCO usually makes improvement. But, BCO gets stuck in some early generations. This fact shows that global evolution in MBCO works quite well to exploit some good solutions in the previous generation. In the early generations, MBCO can produce a *makespan* that close to the global optimum since it uses a special treatment for the best bee.

In the late generations, after 150-th generation, MBCO still makes improvement as it uses dynamic parameters in each generation. This allows MBCO to escape from some stuck evolutions and then reach the expected global optimum. In contrast, BCO that uses static parameters gets stuck and reach a local optimum.

Those facts show that the three proposed modifications work quite well. By exploiting some good solutions to be improved in the next generation keeps some promising partial solution in the colony. Two dynamic parameters, NC and the probability function, keep the diversity of the population. This allows MBCO to escape from some stuck evolutions. The TS performed to the best complete solution leads the colony to reach a solution that closes to the global optimum, even in the early generations.

MBCO is applied to solve 10 JSP instances to see if the three proposed modifications work quite well in general. In this research, MBCO uses the same parameters as in BCO, i.e. the maximum iteration of 300 and the number of bee ranges from 9 to 49. There are four additional parameters used in MBCO, namely: 1) the number of constructive moves (NC), which is random for each generation that ranges from 1 to 10% of the number of operations; 2) the maximum iteration in TS, which ranges from 100 to 300; 3) the number of neighbors generated by TS, which is in the interval of 10 to 20; and 4) the length of tabu list, i.e. 10% of the maximum iteration of TS.

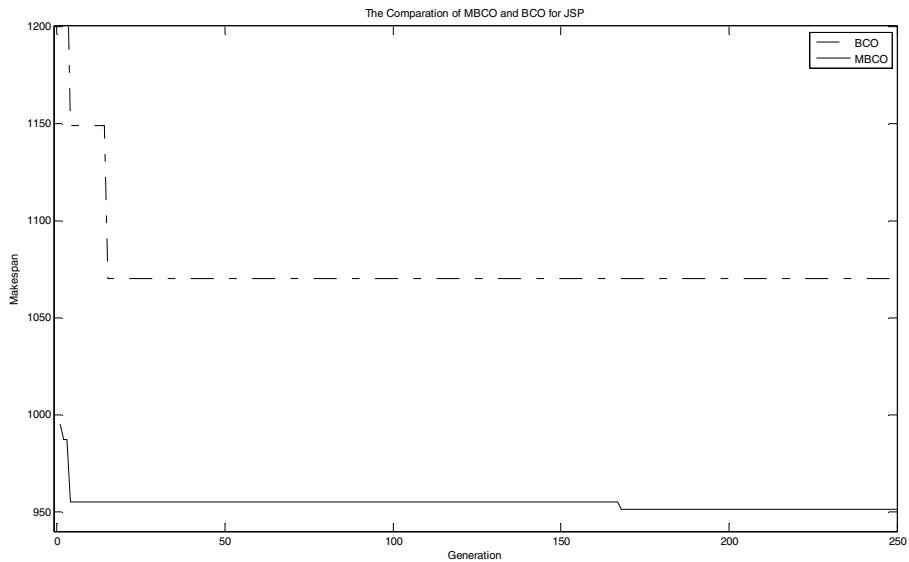


Fig. 6. BCO and MBCO is applied to solve a JSP instance, la09. BCO gets stuck and reach a local optimum with *makespan* of 1070. But, MBCO improves quite faster than BCO in the early generations and reach a global optimum with *makespan* of 951.

Using those parameters, BCO and MBCO are examined using 10 JSP instances. Table 1 shows the worst, best and average accuracy for the instances using BCO and MBCO. The accuracy is calculated by a formula in Equation (3). The table 1 shows that for all JSP instances, MBCO gives better accuracy than BCO. In two instances, ft06 and la09, MBCO found the best solution with accuracy of 100%. However, MBCO is not quite optimal for some complex JSP instances with 200 operations or more, such as abz8, la29, la31, and la37.

$$\text{Accuracy} = \frac{\text{best makespan known}}{\text{best makespan found}} \times 100\%. \quad (3)$$

Table 1. Results of computer simulation of BCO and MBCO, 10 runs for each JSP instance

JSP instances	Best Solution Known	Number of process	BCO Accuracy (%)			MBCO Accuracy (%)		
			Worst	Best	Avg.	Worst	Best	Avg.
abz05	1234	100	78.0	82.3	80.0	83.8	95.5	89.1
abz8	645	300	57.1	60.4	58.2	70.7	74.2	72.0
ft06	55	36	91.7	94.8	93.2	91.7	100	96.2
ft20	1165	100	78.3	81.8	79.9	87.1	97.4	91.0
la03	597	50	83.5	87.2	85.6	85.8	93.6	88.8
la09	951	75	88.1	92.8	90.0	89.4	100	96.7
la21	1046	150	68.2	71.3	70.0	74.1	88.4	82.9
la29	1142	200	62.5	67.7	64.7	69.4	81.3	74.4
la31	1784	300	70.6	74.0	71.7	86.4	87.7	87.2
la37	1397	225	68.2	73.8	70.1	79.2	83.2	81.1

5 Conclusion

Modified BCO significantly gives higher accuracy than the original BCO for the 10 JSP instances studied. The three proposed modifications allow MBCO to evolve quite faster than BCO. But, MBCO is not quite optimal for some complex JSP instances with many operations. An improvement procedure could be performed by exploiting any partial solution to be kept in the next generation, not only the first one.

Acknowledgments. We would like to thank to Mrs. Ema Rachmawati and also our colleagues in Telkom Institute of Technology (IT Telkom) for the advices and motivations.

References

1. Teodorović, D., Dell'orco, M.: Bee Colony Optimization-A Cooperative Learning Approach to Complex Transportation Problems (2010)
2. Sivakumar, I.A., Chong, C.S., Gay, K.L., Low, M.Y.H.: A Bee Colony Optimization Algorithm to Job Shop Scheduling, pp. 1954–1961. IEEE, Los Alamitos (2006) 1- 4244-0501-7/06
3. Geyik, F., Cedimoglu, I.H.: The Strategies and Parameters of Tabu Search for Job Shop Scheduling. Journal of Intelligent Manufacturing 15, 439–448 (2004)
4. Lestan, Z., Brezocnik, M., Buchmeister, B., Brezovnik, S., Balic, J.: Solving The Job-Shop Scheduling Problem With A Simple Genetic Algorithm. Int. J. Simul. Model. 8(4), 197–205 (2009)
5. Teodorović, D.: Bee Colony Optimization BCO. In: Lim, C.P., Jain, L.C., Dehuri, S. (eds.) Innovations in Swarm Intelligence. SCI, vol. 248, pp. 39–60. Springer, Heidelberg (2009) ISBN 3642042244, 9783642042249
6. Teodorović, D., Lučić, P., Marković, G., Dell'orco, M.: Bee Colony Optimization: Principles and Applications. In: 8th Seminar on Neural Network Applications in Electrical Engineering, NEUREL 2006, Belgrade, Serbia (2006)

Polytope Classifier: A Symbolic Knowledge Extraction from Piecewise-Linear Support Vector Machine

Vilen Jumutc¹ and Andrey Bondarenko^{1,2}

¹ Riga Technical University, Meza 1/4, LV-1658 Riga, Latvia
Jumutc@gmail.com

² CTC Co Ltd., Jurkalnes 15/25, LV-1046 Riga, Latvia
Andrejs.Bondarenko@gmail.com

Abstract. This paper describes an extension of a symbolic knowledge extraction approach for Linear Support Vector Machine [1]. The proposed method retrieves a set of concise and interpretable IF-THEN rules from a novel polytope classifier, which can be described as a Piecewise-Linear Support Vector Machine with the successful application for linearly non-separable classification problems. Recent major achievements in rule extraction for kernelized classifiers left some reasonable and unresolved problems in knowledge discovery from nonlinear SVMs. The most comprehensible methods imply constraints that strictly enforce convexity of the searched-through half-space of inducted SVM classifier [2]. Obviously non-convex hyper-surfaces couldn't be effectively described by a finite set of IF-THEN rules without violating bounds of a constrained non-convex area. In this paper we describe two different approaches for "learning" a polytope classifier. One of them uses Multi-Surface Method Tree [3] to generate decision half-spaces, while the other one enables clustering-based decomposition of target classes and initiates a separate Linear SVM for every pair of clusters. We claim that the proposed polytope classifier achieves classification rates comparable to a nonlinear SVM and corresponding rule extraction approach helps to extract better rules from linearly non-separable cases in comparison with decision trees and C4.5 rule extraction algorithm.

1 Introduction

During the past decades the ability to provide explanations of decisions made by rapidly increasing number of classification methods has deserved a reasonable attention due to a high importance in assisting crucial decision makers and domain experts. Therefore decision support is taken for granted in emerging amount of terabytes of unclassified and unsupported information. Unfortunately, the intractable and opaque representation of the decision made by any "black-box" model only complicates its successful application in decision-support.

In this paper we present a novel algorithm for the extraction of rules from a Piecewise-Linear Support Vector Machine. The latter classifier and corresponding rule extraction approach neither extend nor replace the existing algorithms

that solely depend on original or synthetic set of Support Vectors [24], but rather ensures that discovered rules will asymptotically describe the polytope-based classifier obtained as a Piecewise-Linear SVM (P-LSVM) using clustering-based decomposition of target classes. Such decomposition implies that we cluster initial dataset separately for each target class and obtain unique P-LSVM (polytope) for every cluster from selected target class as well.

Further we state a separate optimization problem for each P-LSVM classifier to extract rules of IF-THEN form. To get an asymptotic approximation of every polytope we propose an algorithm for recursive rule extraction. The extracted rules of IF-THEN form can then be easily interpreted by any domain expert. Contingent rules are build of simple conjunctive interval-based expressions where every statement defines a separate hyperplane orthogonal to an axis of dimension being queried. To build a final classifier all found rules are joined together into one decision operator with every statement or hypercube being evaluated in an inclusive disjunction. Moreover each evaluated hypercube is obtained as a separate optimization problem resulted from a recursive partitioning of the polytope classifier. Using this approach we consequently search through uncovered areas of the aforementioned polytope and try to inscribe hypercubes with the largest possible volume.

The rest of the paper is organized as follows: Sections 2 and 3 outline the Soft-Margin Support Vector Machine and Multi-Surface Method Tree, which are employed to create P-LSVM classifier, Section 4 covers in detail the proposed polytope-based classifier and rule extraction method, as well as gives a necessary lemma for convexity of the polytope region. Section 5 describes experimental setup and numerical results, while Section 6 concludes the paper.

2 Support Vector Machine

Support Vector Machine is based on the concept of separating hyperplanes that define decision boundaries using Statistical Learning Theory [5]. Using a kernel function, SVM is an effective and robust classifier in which the optimal solution or decision surface is found by solving the quadratic programming problem with linear constraints, rather than by solving a non-convex, unconstrained minimization problem as stated in typical back-propagation neural network.

Further we present only dual representation of the Soft-Margin SVM's primal objective that is given in terms of its Lagrangian multipliers - λ_i and can be effectively optimized using any off-the-shell linear optimizer that supports constraint adaptation:

$$\max_{\lambda} \left\{ \sum_{i=1}^l \lambda_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \lambda_i \lambda_j y_i y_j K(x_i, x_j) \right\}, \quad C \geq \lambda_i \geq 0, \quad \sum_{i=1}^l \lambda_i y_i = 0, \quad (1)$$

where C represents regularization constant, λ_i represents a Lagrangian multiplier, y_i is $\{\pm 1\}$ - valued label of a data sample x_i , $K(x_i, x_j)$ is a kernel function and l is the number of training samples.

Final classification of a new sample x' is derived by: $d = \text{sign}(\sum_i \lambda_i K(x_i, x') + b)$, where $K(x_i, x')$ and b correspond to a kernel function evaluated for a new sample and a linear offset of the optimal decision hyperplane.

3 Multi-Surface Method Tree

Multi-Surface Method Tree [3] utilizes linear programming to build classification tree which is capable of correct classification of linearly non-separable target classes. As described in [8] the algorithm deals with two sets of points: \mathcal{A} and \mathcal{B} in the n -dimensional euclidean space R^n . LP problem to find separating hyperplane is stated as follows:

$$\begin{aligned} & \min_{w, \gamma, y, z} \left\{ \frac{1}{m} ey + \frac{1}{k} ez \right\}, \\ & \text{s.t. } y \geq -Aw + e\gamma + e, \\ & \quad z \geq Bw - e\gamma + e, \\ & \quad y \geq 0, \\ & \quad z \geq 0, \end{aligned} \tag{2}$$

where w and γ represent a classification hyperplane, y and z introduce corresponding error vectors for both target classes, e is a vector of ones and m, n correspond to the number of samples in two sets of points: \mathcal{A} and \mathcal{B} .

Linear classifier which can "strictly" separate aforementioned sets \mathcal{A} and \mathcal{B} can be found by solving inequalities: $Aw > e(\gamma + 1)$, $Bw < e(\gamma - 1)$, here both A and B are $m \times n$ and $k \times n$ matrices respectively representing datasets \mathcal{A} and \mathcal{B} holding n -dimensional vectors. w is n -dimensional "weight" vector representing the normal to the "optimal" separating hyperplane, and γ is a real number which is a "threshold" that locates discrimination hyperplane $wx = \gamma$. The weights $\frac{1}{m}$ and $\frac{1}{k}$ ensure that nontrivial w vector is always generated without imposing any extraneous constraints.

The described approach [8] is called Multi-Surface Method Tree, because it is an extension of a multi-surface method for pattern recognition with decision trees. For each node in the tree, the best split of training points reaching that node is found by solving the stated LP problem using any suitable optimizer. In the scope of ongoing Section [1] it should be noted that generated by MSM-T intersecting half-spaces immediately imply polytope classifier which is a cornerstone of the proposed rule extraction approach.

4 Proposed Method

In this section we introduce a specific extension of a widely acknowledged rule extraction algorithm for Linear Support Vector Machine [1]. Firstly we present an approach for obtaining a clustering-based decomposition of target classes and acquisition of corresponding P-LSVM classifier. Secondly we present a constrained linear programming problem for approximation of every such classifier (and corresponding polytope) with a hypercube with the largest possible volume. And finally we evolve an iterating and recursive algorithm for obtaining

the most extensive coverage of every polytope by the finite set of inducted rules (hypercubes).

4.1 Clustering-Based Decomposition and Polytope Classifier

To obtain corresponding clustering-based decomposition of target classes, we apply very simple clustering algorithm, namely K-Means [6] that allows very quick and intuitive partitioning of any dataset into a fixed number of spherical clusters. After running K-Means we iterate over the clusters of only one selected target class and initiate a different Linear SVM for each pair of clusters (namely for one of the current target class and for one from another). This general formulation results in n -dimensional convex area given as a set of n -dimensional points describing an implied polytope(s):

$$\begin{aligned} \{x \in \mathbb{R}^n | \langle w_i, x \rangle - b_i \geq 0, \\ s.t. \quad w \in \mathbb{R}^n, b \in \mathbb{R}, \forall i \in \mathbb{N} \wedge i \leq m\}, \end{aligned} \quad (3)$$

where m is the number of obtained SVM decision hyperplanes given by $\langle w_i, x \rangle = b_i$ term. This formulation implies that the resulting area or polytope is a convex set of points and is defined as an intersection of the finite number of half-spaces inducted by aforementioned SVM decision hyperplanes. The latter proposition is proven by the following lemma:

Lemma 1. *Given the n -dimensional polytope (area) R resulted from the intersection of several SVM decision hyperplanes and inducted half-spaces $H_1 \cap H_2 \cap \dots \cap H_m$ provides a convex set of points belonging to an initial training sample.*

Proof. The inducted by Linear SVM decision half-space is already convex itself and a finite number of such convex half-spaces imply a unique H-description of a polytope (area) given by the set of facet-defining half-spaces [7].

Using the above proposition we infer the P-LSVM classifier which classifies data as follows:

$$\wedge_{i=1}^m (\langle w_i, x' \rangle - b_i \geq 0) = \begin{cases} \text{true}, x' \in A_+, \\ \text{false}, x' \in A_-, \end{cases} \quad (4)$$

where x' is a test example, A_+ , A_- denote positive and negative target classes and m corresponds to a unique SVM employed in the P-LSVM classifier.

4.2 Optimization Problem

In the previous section, we described an approach for clustering-based decomposition of target classes and definition of P-LSVM classifier for each unique cluster.

We reformulate every such classifier as a single rule of the form: $\wedge_{i=1}^m (\langle w_i, x' \rangle - b_i \geq 0)$. Now we could see that P-LSVM classifier forces very suitable constraints for the linear programming problem where we deal with hypercubes rather than hyperplanes as described in Eq 5.

This section presents a linear programming problem in a manner very similar to [1], where authors maximized the volume of a bounded hypercube inscribed into SVM-induced half-space. The actual volume of any hypercube is given by a product of all unique edges and could be represented by the following term: $\prod_{i=1}^n |u_i - l_i|$, where u_i is an upper bound and l_i - a lower bound for the i -th hypercube's dimension. Instead of a quite inappropriate for solving "product"-based representation we could effectively replace one by taking a natural logarithm and substituting an aforementioned product by the sum of $\log |u_i - l_i|$. We present our problem by the following constrained optimization criteria:

$$\begin{aligned} & \max_{u,l} \sum_{i=1}^n \log(u_i - l_i), \\ & \text{s.t. } u_i \geq l_i \quad \forall i, \\ & \quad \langle w_j, l \rangle - b_j \geq 0 \quad \forall j, \\ & \quad \langle w_j, u \rangle - b_j \geq 0 \quad \forall j, \\ & \quad \langle w_j, l \rangle \geq -\infty \quad \forall j, \\ & \quad \langle w_j, u \rangle \leq \infty \quad \forall j, \end{aligned} \tag{5}$$

where w_j and b_j correspond to a norm-vector and an offset of the j -th SVM decision hyperplane, l and u are given in the vector form, are indexed by i and represent upper and lower bounds on hypercube and \log is a natural logarithm.

It should be noted that akin it was stated in [1] instead of a maximum volume criteria one can use point coverage maximization criteria to define a different optimization problem.

4.3 Generalized Rule Extraction Algorithm

The solution of the stated above optimization problem (Eq 5) defines two vertices, namely representing the *lower* and *upper* bounds of the found hypercube. Having a single hypercube inscribed into the polytope can be insufficient in terms of classification fidelity. To overcome this undesirable result we can search through for additional hypercubes inscribed into the remaining regions of the induced polytope area. This process could be repeated recursively so that ongoing search for the smaller uncovered regions will generate more and more rules that will asymptotically approximate original P-LSVM classifier with the desired level of classification fidelity.

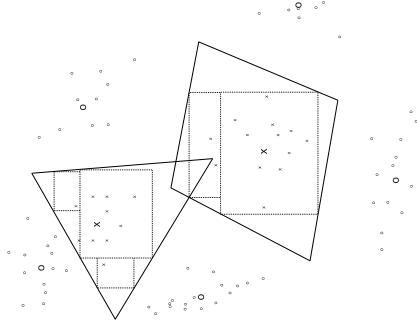


Fig. 1. Polytopes and rules (rectangles) separating x and o cluster centers of the x and o target classes

Such remaining regions of interest could be defined as follows:

$$\begin{aligned} I_i^l &= \left\{ x \in R^n, \text{ s.t. } \begin{array}{l} l_j^* < x_j \leq u_j^* \quad \forall 1 \leq j \leq i, \\ x_i \leq l_i^*, \end{array} \right. \\ I_i^u &= \left\{ x \in R^n, \text{ s.t. } \begin{array}{l} l_j^* \leq x_j < u_j^* \quad \forall 1 \leq j \leq i, \\ x_i \geq u_i^*. \end{array} \right. \end{aligned} \quad (6)$$

Here I_i^l and I_i^u are polytope regions surrounding the extracted rule for the i -th dimension, l and u are given in the vector form and represent the lower and upper bounds (vertices) of the currently processed hypercube (rule). Presented in Eq. 6 rule inequalities are satisfied for the first $i - 1$ dimensions of x , the inequality that relates to the i -th dimension is not satisfied, and the rest dimensions are free and shouldn't be constrained. Due to the reason that we are dealing with the extracted rule and its surrounding polytope regions for the ongoing search of the fine-grained rules it is obligate to assure that the regions given above don't intersect and all of the rules differ in terms of presented coverage and classification value. Consider dimensions i, j with $j > i$. For each $x \in I_j$, we have $l_i^* < x_i < u_i^*$ and for each $x \in I_i$, we have $x_i \leq l_i^*$ or $x_i \geq u_i^*$. Hence, I_i are non-intersecting, and the rules that we arrive at for each I_i differ in terms of approximated polytope region.

Figure 1 shows objects that are operated by Algorithm 1 Polytopes (P-LSVMs) classify x target class and make use of corresponding clustering-based decomposition. Rules are represented by rectangles. Largest rectangle inscribed into each polytope is a corresponding rule extracted with the recursion depth $d = 1$, smaller rectangles are found using deeper recursion steps.

Algorithm 1 depicts symbolic rule extraction routine from a single polytope presented by *hyperplanes* variable. Thus such routine should be separately applied to every cluster for which P-LSVM classifier is inducted. Variable I_{region}

holds boundaries across all dimensions and defines a working domain. Initially I_{region} is initialized to $[-\infty, +\infty]$ across all dimensions, but for deeper recursion steps it is initialized to subspaces targeted for more "fine-grained" rule extraction, as I_{region} itself participates in defining subsequent search regions. d_{max} denotes maximum recursion depth, variable R holds for extracted rules. Variables l^* and u^* are n -dimensional vectors denoting inscribed hypercube with "lower" and "upper" vertices uniquely defining extracted IF-THEN rule. C holds for indexes of points covered by already extracted rules present in R and U holds for indexes that aren't currently covered by any rule.

Algorithm 1. Rule Extraction Algorithm

```

input : set of hyperplanes hyperplanes, working region  $I_{region}$ ,
        maximal recursion depth  $d_{max}$ , initially empty set of rules  $R$ 
output: set of classification rules  $R$ 
1 ExtractRules( $I_{region}$ ,  $d$ ,  $R$ )
2   if  $d > d_{max}$  then
3     | return
4   end
5    $lb \leftarrow \text{GetLowerBound}(I_{region})$ 
6    $ub \leftarrow \text{GetUpperBound}(I_{region})$ 
7    $[l^*, u^*] \leftarrow \text{Optimize}(\text{hyperplanes}, lb, ub)$ 
8    $R_{new} = \text{Rule}(l^*, u^*)$ 
9    $R \leftarrow R \cup R_{new}$ 
10   $C \leftarrow \text{GetCoveredSamples}(R)$ 
11   $U \leftarrow \text{GetUncoveredSamples}(R, C)$ 
12  if IsEmpty( $U$ ) then
13    | return
14  end
15   $I_{subRegions} \leftarrow \text{GetSurroundingRegions}(l^*, u^*)$ 
16  foreach  $I_i$  in  $I_{subRegions}$  do
17    | if CountCoveredSamples( $I_i, U$ )  $> 0$  then
18      |   |  $R \leftarrow \text{ExtractRules}(I_i, d + 1, R)$ 
19    |   end
20  | end
21 end

```

It should be noted that polytopes could have intersections between each other, thus extracted rules (hypercubes) could be intersecting as well. Optimization of extracted rules, removal of overlapping rules or boosting of the resulted set of rules is beyond the scope of the current paper. In experimental part we provide only averaged number of all extracted rules across 10 independent runs. The generalized rule extraction algorithm is comprehensively described in Algorithm **II**.

5 Experimental Part

To verify and test our rule extraction approach we have selected several public UCI datasets [9]. Monks-1 and Monks-2 are quite known and highly referenced benchmark datasets in the rule extraction literature while "Balance-Scale" represents very simple psychological experiment and is very suitable for the concise and interpretable rule extraction. Another verified dataset was "Normally Distributed Clustered Datasets on Cubes" [10], which was generated using the following control parameters: $dimensions = 3$, $spread = 100$, $points = 300$, where variables $dimensions$ and $points$ are self explanatory, while $spread$ controls dispersion of points located around the vertices of cubes. The induced set of parameters provides highly non-homogeneous and linearly non-separable target classes. Additionally we should describe "Balance-scale" dataset where we have selected only two target classes, namely L and R , classifying whether a person is a left-balanced or right-balanced writer.

During our experiments we have faced a drastic classification rate fluctuations that were tightly bounded to an initial number of clusters for different target classes passed into the K-Means algorithm. The number of clusters for both classes was determined heuristically using training datasets for Monks problems and ten-fold cross validation for NDCC and "Balance-Scale" datasets. Optimal number of clusters was more or less stable in terms of higher classification accuracy on training data, although there is no any guarantee that optimality determined for the training data will be the same for verification dataset. Additionally we provide averaged number of all extracted rules and execution times across 10 independent trials. The aforementioned indicators and parameters of our approach are presented in Table II

Table 1. P-LSVM indicators & parameters

Dataset	Number of clusters	Number of rules	Execution time(sec)
Monks-1	4	214±0	14.358±0.882
Monks-2	60	7437.1±1700.5	2244.718±539.715
NDCC	60	5140.4±377.76	253.854±19.469
Balance-scale	60	9392.1±422.43	896.439±44.463

For all datasets we have used recursion depth in Algorithm II - d_{max} equal to 5. Results of a classification accuracy for benchmark SVM methods, other rule-based classifiers as well as aforementioned rule extraction approach (given by Algorithm II) are presented in Table II. To verify and test our proposed rule extraction approach we evaluated UCI datasets under the following experimental setup: for datasets that weren't originally separated into validation and training sets we performed 10-fold cross-validation and collected averaged classification accuracy. For others we tested our approach on presented in UCI repository validation set and collected single classification accuracy rate. For NDCC dataset we evaluated our approach over 10 independent trials and collected mean classification accuracy. To collect indicators in Table II we additionally performed

10 independent trials for Monks-1 and Monks-2 datasets. All experiments were done using following hardware setup: Core 2 Duo processor, 4GB RAM and Linux OS.

We compare our approach with the widely acknowledged Linear SVM and RBF-SVM. For all datasets we have selected the fixed C regularization parameter and enclosed subspace for γ parameter of RBF kernel with some initial guess of its corresponding scaling factor¹ while applying Multiple Kernel Learning [11] approach to determine its optimal value.

As it can be seen from Table 2 all problems except "Balance-Scale" are poorly separable using Linear SVM. On the other hand, approximation of a nonlinear decision surface gives a necessary boost of the classification accuracy for polytope classifier and for extracted rules. Apparently we can see that rules extracted from the P-LSVM classifier induced by clustering decomposition are superior to the ones extracted from polytopes created by MSM-T. The latter could be explained by the nature of the stated LP problem. To achieve comparable with Piecewise-Linear SVM results, we need to drastically increase recursion depth of Algorithm 1. Additionally we could see from Table 1 that our method generates quite enlarged number of rules within some reasonable time for noisy and hardly discriminated target classes. But this result is a rough approximation of underlying polytopes that could aggregate data points very close to separating hyperplanes thus enabling very high recursion depth and number of inducted rules. We are absolutely confident that usage of appropriate pruning and boosting strategies could help us to significantly reduce number of classification rules.

Table 2. Classification accuracy (%)

	Classifier	Monks-1	Monks-2	NDCC	Balance
SVM _{linear}	65.509	67.130	73.333	93.730	
SVM _{rbf}	86.806	80.556	95.000	98.082	
C4.5	75.690	65.050	74.000	70.780	
MSM-T	83.565	79.630	88.667	87.526	
Rules _{msm-t}	69.444	54.861	75.333	68.524	
P-LSVM	99.537	80.324	93.667	97.913	
Rules _{p-lsvm}	96.296	74.537	89.738	96.857	

6 Conclusion

In this paper we described a novel rule extraction approach from a novel P-LSVM classifier using clustering-based decomposition of target classes. We have found that the polytope-based classifier and inducted rules are sensitive to K-Means clustering and corresponding initial number k of spherical clusters. Along with

¹ We have defined range of $b_\gamma \cdot 10^{[-10...10]}$ with the step 0.25 resulting in a total of 81 γ -parameters where b_γ is a corresponding scaling factor of γ stated as follows: $b_\gamma = 1/2 \cdot \sqrt{\text{median}(X)}$ where X is a vector of all dataset values.

clustering-based decomposition of target classes we showed that Algorithm 1 for rule extraction could be applied to MSM-T [3] classifier but without a deeper recursion we couldn't increase classification rates.

Finally observing Eq. 5 and Algorithm 1 we could clearly see that our approach is very computationally intensive in terms of the number of constraints being queried. Obviously that another very important topic is an optimization of intersecting rules and necessity for appropriate pruning strategy as we could see from Table 1. The provided experimental results verify that the P-LSVM classifier and proposed method for rule extraction is capable of performing rule-based classification in reasonable time better than the C4.5 algorithm, MSM-T and for some datasets performs even better than a benchmark nonlinear SVM.

References

1. Fung, G., Sandilya, S., Bharat Rao, R.: Rule Extraction from Linear Support Vector Machines. In: Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, Illinois, USA, April 21–24, pp. 32–40 (2005)
2. Ren, L., Garcez, A.: Symbolic Knowledge Extraction from Support Vector Machines: A Geometric Approach. In: Köppen, M., Kasabov, N., Coghill, G. (eds.) ICONIP 2008. LNCS, vol. 5507, pp. 335–343. Springer, Heidelberg (2009)
3. Mangasarian, O.-L.: Mathematical Programming in Neural Networks. ORSA Journal on Computing 5(4), 349–360 (1993)
4. Nunez, H., Angulo, C., Catala, A.: Rule extraction from support vector machines. In: Proceedings of the 10th European Symposium on Artificial Neural Networks, Bruges, Belgium, April 24–26, pp. 107–112 (2002)
5. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, Heidelberg (1995)
6. Lloyd, S.-P.: Least squares quantization in PCM. IEEE Transactions on Information Theory 28(2), 129–137 (1982)
7. Grunbaum B.: Convex Polytopes, 2nd edn., prepared by Kaibel, V., Klee, V., Ziegler, G.M. (2003)
8. Bennet K.-P.: Desicion Tree Construction via Linear Programming. In: Proceedings of the 4th Midwest Artificial Intelligence and Cognitive Science Society Conference, pp. 97–101 (1992)
9. Frank, A., Asuncion, A.: UCI Machine Learning Repository, School of Information and Computer Science, Irvine, University of California, USA (2010), <http://archive.ics.uci.edu/ml>
10. Thompson M.-E.: NDCC: Normally Distributed Clustered Datasets on Cubes, Computer Sciences Department, University of Wisconsin, Madison, USA (2010), <http://www.cs.wisc.edu/dmi/svm/ndcc/>
11. Rakotomamonjy, A., et al.: SimpleMKL. Journal of Machine Learning Research 9, 2491–2521 (2008)

A Library of Nonlinearities for Modeling and Simulation of Hybrid Systems

Florin Ionescu^{1,2}, Dragos Arotaritei³, Stefan Arghir⁴, George Constantin⁴,
Dan Stefanoiu⁴, and Florin Stratulat⁴

¹ HTWG-Konstanz, D-78462 Konstanz, Germany

² Steinbeis Transfer Institute Dynamic Systems, D-10247 Berlin, Germany

³ University of Medicine and Pharmacy "Gr. T. Popa", RO-700115 Iasi

⁴ University "Politehnica" of Bucharest RO- 060042 Bucharest, Romania

florin.ionescu@stw.de,

{ionescu,darotari,sarghir,geo}@htwg-konstanz.de

Abstract. The present paper is a short review of some of the concepts, procedures and achieved results, oriented toward the modeling of complex, large installations, known under the name of HYPAS. Physical phenomena involved in hydraulic, pneumatic, mechanic and electric systems are depicted by mathematical models (MM) with static and dynamic nonlinearities (NL). This means that, when building large installations, the MM provide multiple NL. They should be completely known in order for the controller to be designed appropriately enough to provide desired behavior, while complying with economical conditions. To facilitate the automatic mathematical model generation, a library with multilayer models was created and are made available. This approach was necessary because the MATLAB® library provides only few general NL. In cases where applicable, neuro-fuzzy techniques were employed for modeling of NL. The paper presents some of the most important results, accompanied by their MATLAB Simulink® representation.

Keywords: Hybrid Systems, Nonlinear Systems, Modeling and Simulation, Libraries, HYPAS.

1 Introduction

Most mathematical models of pneumatic, hydraulic and electric systems are nonlinear. Nonlinear differential equations describe their behavior and this also applies to hybrid systems, *e.g.* servo systems. Principle of superposition does not apply for these systems ([3], [18], [19], [21]). Among the nonlinear systems, the hydraulic ones are, maybe, the most nonlinear. Various mechanical, hydraulic and electrical phenomena are contributing to this situation. Nonlinearities can be statistical, dynamical, structural or time dependent. They are responsible for critical situations occurring during operation. Also they are hindering theoretical investigations and the numerical simulations. To simulate a hydraulic drive system, the critical problem is to possess an

accurate mathematical model. Thus, the modeling of nonlinearities is a *sine qua non* task in order to accurately describe dynamic processes ([1], [2], [17], [21]).

Some of these NL along with the respective mathematical equations for analytical models are given in Table 1. We can see from this table that many systems have hysteretic behavior. Other frequent differential equation models have solutions given as nonlinear piecewise monotonic functions.

Table 1. Some of the studied NL, described with mathematical graph, mechanical example, mathematical function and Matlab® figure

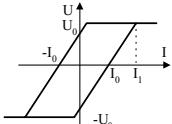
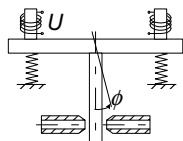
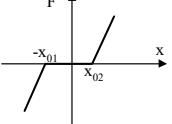
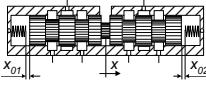
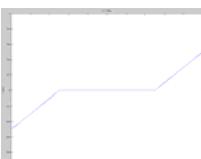
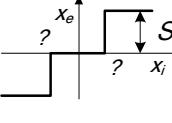
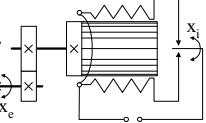
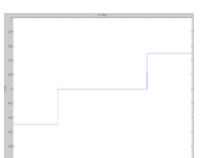
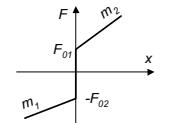
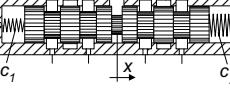
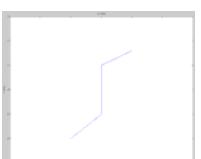
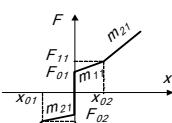
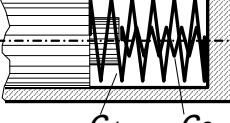
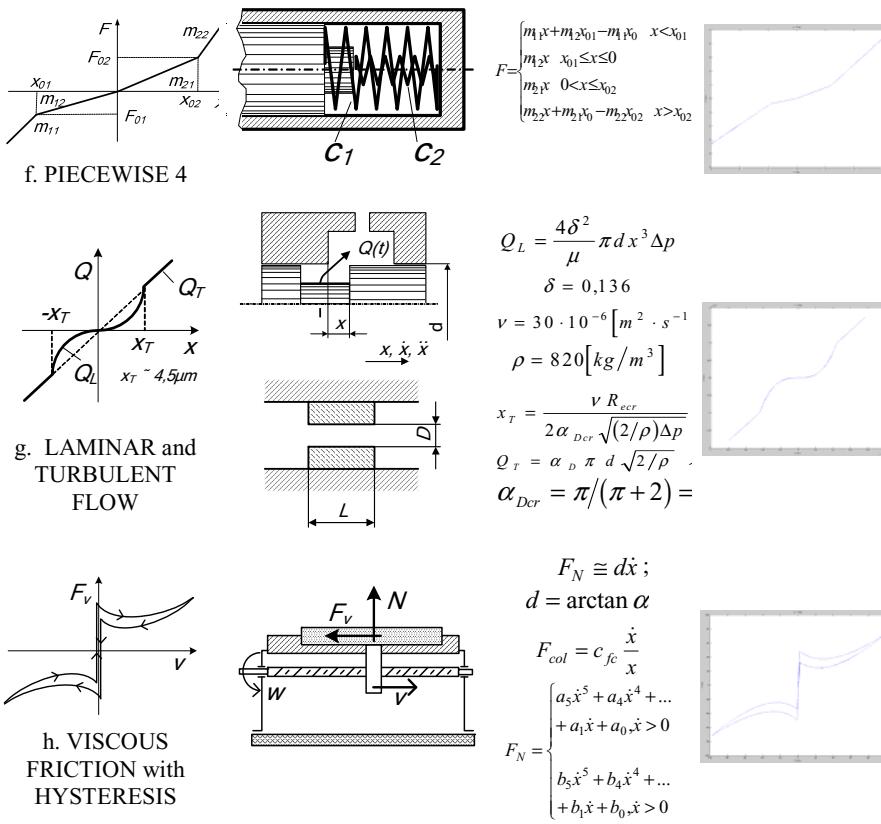
Graphical representation	Mechanical representation	Mathematical expression	Matlab® figure
 a. HYSTERESIS		$U = I - I_0, \quad \frac{dI}{dt} > 0$ $U = I + I_0, \quad \frac{dI}{dt} < 0$ $\frac{dU}{dI} = 0, \quad U - I < I_0$	
 b. INSENSIBILITY ZONE 1 st order		$F = \begin{cases} 0 & x \in [-x_01, x_02] \\ mx - F_0 & x > x_02 \\ mx + F_0 & x < -x_01 \end{cases}$	
 c. INSENSIBILITY ZONE 2 nd order		$x_e = \begin{cases} 0 & x_i < \eta \\ s x_i & s x_i \geq \eta \\ -s x_i & -s x_i \leq -\eta \end{cases}$	
 d. DOUBLE SLOPE		$F = \begin{cases} 0, & x = 0 \\ F_{01} + m_2 x, & x > 0 \\ -F_{02} - m_1 x, & x < 0 \end{cases}$	
 e. TRIMODULAR		$F = \begin{cases} c_1 x, & x < x_{02} \\ c_1 x_{02} + (c_1 + c_2) x, & x > x_{02} \end{cases}$	

Table 1. (Continued)

In controlled systems, hysteresis can have a series of undesirable effects, including loss of robust stability, limit cycles and steady-state error, to name but a few ([4] – [8]). The major hurdles control engineers must overcome when faced with hysteretic nonlinearities are obtaining an accurate model of the hysteretic behavior and finding corresponding means of analysis and design capable of dealing with nonlinear and non-single-valued behavior.

MATLAB/Simulink [20] is a standard tool for simulation. Simulink uses numerical solvers in order to simulate systems given by differential equations. The actual toolbox, entitled “Discontinuities” include many of the most common discontinuities used in practice. However, some of the discontinuities that are particularly interesting in hydraulics and electrics are nonlinear and they are not included in this toolbox. Moreover, some of them are very complicated to implement. In case we need some of these nonlinearities to be in the same diagram or we frequently use a particular set of them, a toolbox contains all possibilities is very useful.

More complex phenomena, such as friction characteristics, must be carefully analyzed case by case. The modeling environment must deal with these problems in

special ways, since they strongly influence the numerical behavior of the underlying differential equation solver ([9], [10], [16]). Simulink does that for discontinuities from its library. Using these discontinuities and the other modules from existent toolboxes we can develop a new toolbox that includes the proposed set of nonlinearities.

2 Modeling Static and Dynamic Nonlinearities

We study two types of nonlinearities: static nonlinearities and dynamic nonlinearities. A static model doesn't take into account the time, it is a simple mapping of the input to an output. Static models are usually represented by analytic formulas that map the input to an output, using often the word if in order to select different singularity or angular points. The dynamic model depends both on the input and the time. Dynamic models are usually implemented by difference equation or differential equations ([11] - [14]).

One of the most common situations is the construction of one SIMULINK block that must implement the function:

$$y = \begin{cases} f(x) & \text{if } Cond \\ g(x) & \text{otherwise} \end{cases} \quad (1)$$

In the equation above, *Cond* implies the inequality or equality operator. We denote $C=\{1,0\}$ the truth value of the condition *Cond*. Simulink has blocks that implement the functions \geq , $>$, $<$, and \leq . The nonlinearity above can be implemented using SIMULINK blocks if we translate:

$$y = C \cdot f(x) + (1-C) \cdot g(x)) \quad (2)$$

One of the most representative static nonlinearity that can be modelled by the method described above is the Piecewise 4.

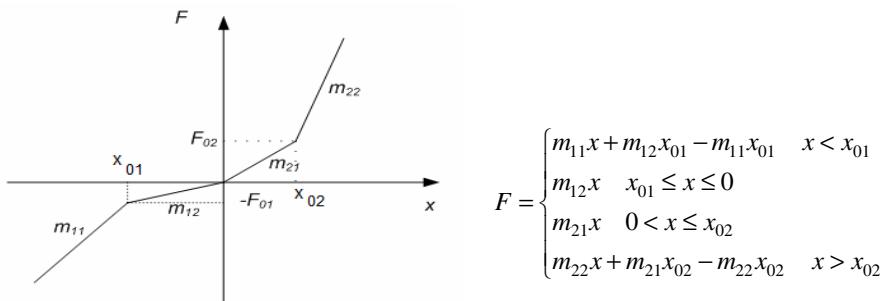


Fig. 1. The nonlinearity Piecewise 4 and its analytical formula

The translation of formula F into an “if-form” is given in the following representation and equations, using the (1), (2).

```

if   (x < x01) // C1
    g1(x) = y = m11x + m12x01 - m11x01
elseif (x01 ≤ x ≤ 0) // C2
    g2(x) = y = m12x
elseif (0 < x ≤ x02) // C3
    g3(x) = y = m21x
else
    g4(x) = y = m22x + m21x02 - m22x02
end

```

$$y = C1 \cdot g1(x) + (1 - C) \cdot G_a(x), \quad C : (x < x_{01}) \quad (4)$$

$$\begin{aligned} G_a(x) &= C2 \cdot g2(x) + (1 - C2) \cdot G_b(x), \quad C2 : (x_{01} \leq x \text{ AND } x \leq 0) \\ G_b(x) &= C3 \cdot g3(x) + (1 - C3) \cdot g4(x), \quad C3 : (0 < x \text{ AND } x \leq x_{02}) \end{aligned} \quad (5)$$

The procedure can be useful in case we want to create a model for static nonlinearities using a mathematic neuron. The function simulates the conditional term that can be expressed as an Hopfield neuron, while the other operation can be simulated using different computational neurons that exist in the Neural Networks toolbox of the Simulink library. Figures 3-6 present the diagrams that were used to model this nonlinearity using the described method.

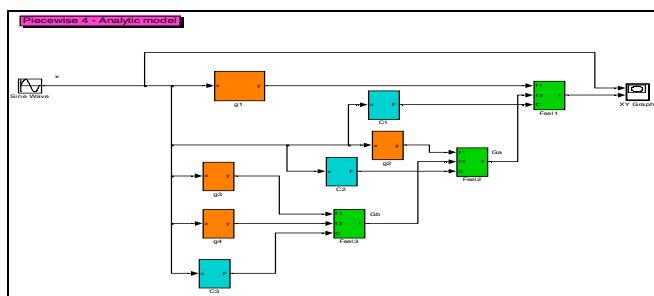


Fig. 3. The nonlinearity Piecewise 4 in Simulink® according to (4)

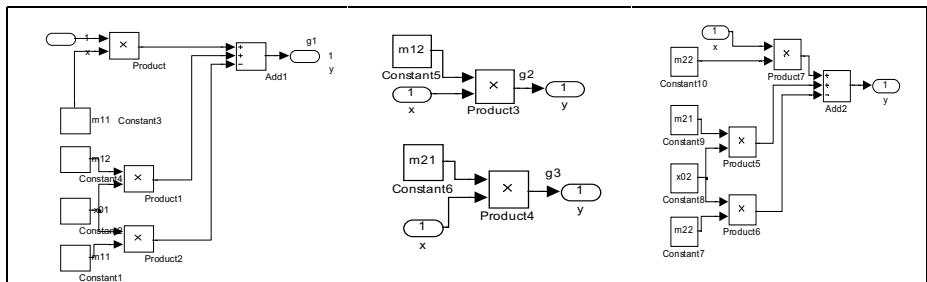
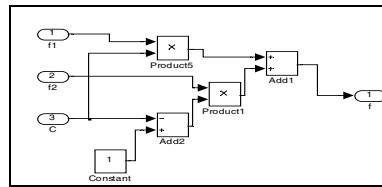
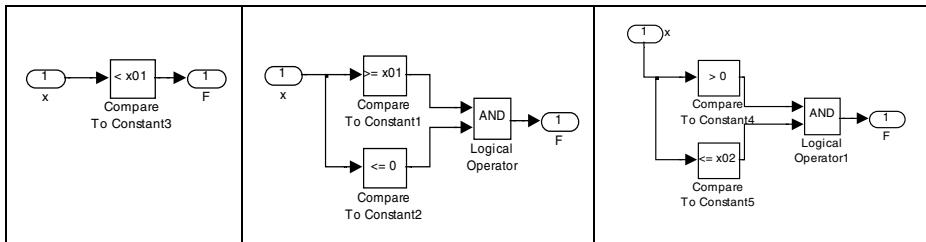


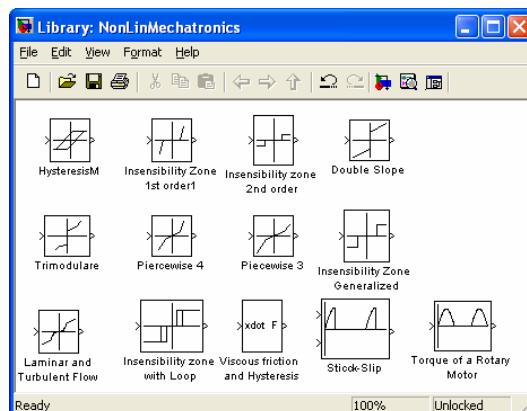
Fig. 4. The subsystems. Block g1(left), g2 & g3 (midle), g4 (right).

**Fig. 5.** The selector Block**Fig. 6.** Conditional blocks

The major disadvantage of the above method is the increased complexity for large numbers of linear pieces. But in practice, this number is limited to maximum 5-6 linear areas, so this approach is feasible.

3 The Library of Nonlinearities

Note that all the new nonlinearities except the hysteretic ones should implement a behavior very similar with the discontinuities already present in the Simulink toolbox. Next, we present all the blocks implemented in the library *NonLinMechatronics*.

**Fig. 7.** The library of nonlinearities

All the blocks have an icon drawn according to the function represented and the input parameters. The blocks are implemented as subsystems with mask. The mask is made according to specification from MATLAB®/Simulink®.

All the blocks are built using elementary Simulink® modules from the standard library. In order to optimize nonlinearities, each one is treated separately, using particular optimization and adequate methods.

An example is the NL Viscous Friction Hysteresis (Fig. 8-9).

The block requires reading the coefficients for both curves, as described in Table 1. They must be present in the MATLAB workspace, having predefined names.

The coefficients can be obtained using polyfit function from MATLAB or in a Simulink scheme using the least square polynomial fit block Polyfit from the Math Functions / Polynomial Functions toolbox.

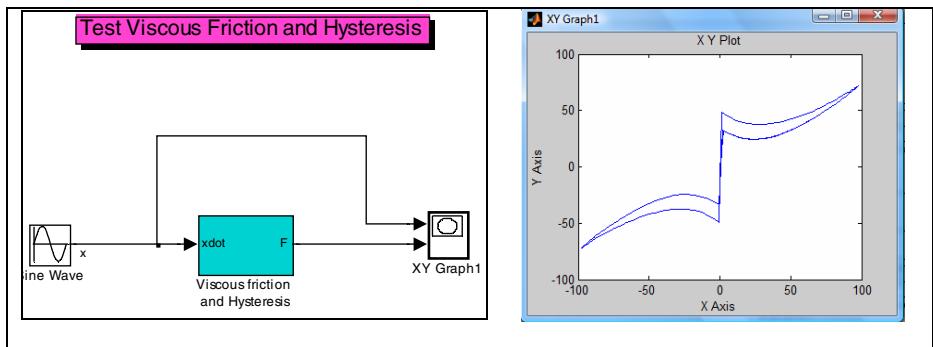


Fig. 8-9. The schema for test Viscous friction and hysteresis, AND The output for test Viscous friction and hysteresis

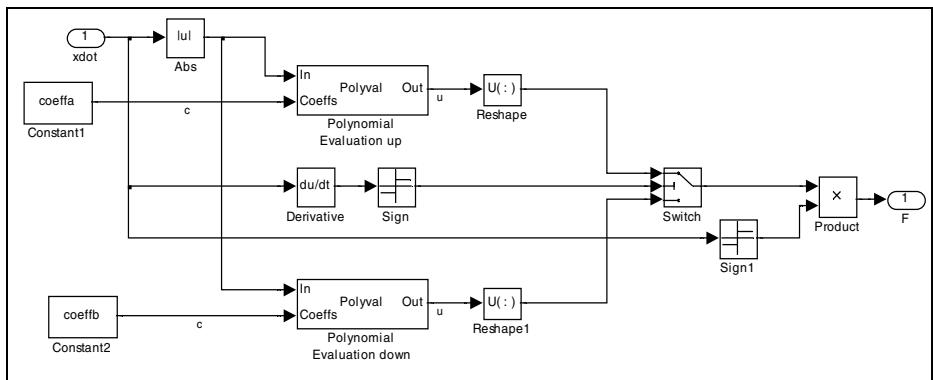


Fig. 10. The subsystem used in Viscous Friction and Hysteresis

The detailed implemented block diagram is presented in Fig. 10. All the other twelve blocks are implemented and tested before prior to inclusion in the library. In order to work with this library, it must be loaded as a Simulink library in the MATLAB environment.

4 A General Method to Construct Piece Wise Nonlinearities Using Neuro-Fuzzy Approaches

In general, neural networks cannot match nonlinear systems exactly. The identification procedure applied to any nonlinear (or linear) system approximate better or worse the plant that exhibits the nonlinear behavior.

In order to use ANFIS (Adaptive Neural Fuzzy Inference System) for nonlinear system identification, the first thing we need to do is to select the input variables. Let us denote by y the output and u the input. In the case of a dynamic system (recurrent ANFIS), we can select the best input candidate from either of following two sets, $Y = \{y(k-1), y(k-2), \dots, y(k-n)\}$ and $U = \{u(k-1), u(k-2), \dots, u(k-m)\}$

For linear piecewise nonlinearities, a general approach can be made using the above observations. The ANFIS parameters are optimized in order to minimize the quadratic error for all the items in the learning stage. The performance indicator is considered to be the maximum of absolute residual values for training stages and the test stage.

For the first nonlinearity, the insensibility zone, the problem to model this nonlinearity by neuro-fuzzy system is that this nonlinearity is in practice a mapping of one input function to an output function, but this mapping must not depend of time. Practically, the input function $f(x)$ can have any time evolution, the ANFIS must learn to create a mapping from $x=f(t)$ to $y=g(t)$, so that we can have a mapping from x to $y(x)$, independent of time or discrete step.

This is a very difficult task for any neuro-fuzzy system: how to solve an analytical problem by a non-analytic system. The training stage is the trick that could solve the problem. We try to teach ANFIS the curve $y(x)$, with a training set chosen by random values, with the corresponding desired output.

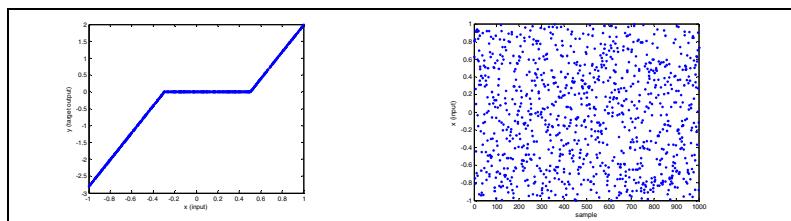


Fig. 11-12. Input-output mapping using neuro-fuzzy AND The set of random points used (both for insensibility zone 1st order)

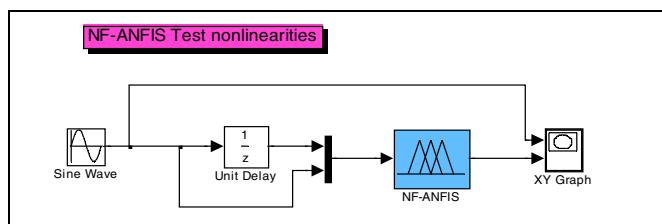


Fig. 13. The general Simulink® diagram for nonlinearities of type: IZ1 (insensibility zone 1st order)

The uniform distribution of these random values ensures, when possible, that the neuro-fuzzy structure learns with the same precision (same error) the exact analytic nonlinearity. The experimental results for the 1st order insensibility zone are shown in Fig 11. The calculated residual error is bellow 2.9×10^{-4} . Nevertheless, there are some nonlinearities that are not that easy model using this approach. For example, the case of the generalized insensibility zone produced a residual error $R_a = 0.229$, as can be seen in Figure 14a. This suggested the use of a different approach, using adaptive recurrent neuro-fuzzy networks (RFNN).

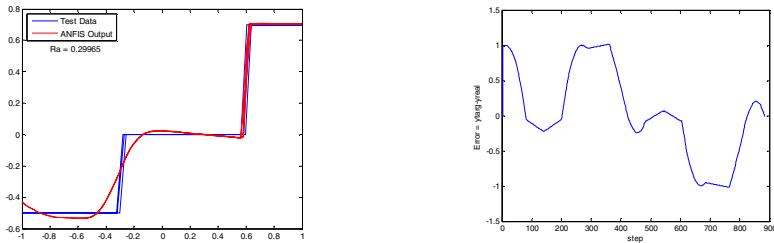


Fig. 14. Results for a) the generalized insensibility zone and b) hysteresis

5 Conclusions

We proposed a new library implemented for Simulink applications for mathematical modeling of pneumatic, hydraulic and electric systems. There are 13 systems in this library that provide a general test diagram for each nonlinearity.

A general method for constructing piecewise nonlinearities is also proposed. The advantages and disadvantages of the proposed method are discussed.

Finally, a novel approach to modeling piecewise nonlinearities using neuro-fuzzy systems is proposed.

In the future, we plan to append other nonlinearities to this library. Also, we plan to develop a recurrent neuro-fuzzy structure, implemented in Simulink that can be used for complex nonlinearities, e.g. hysteresis form.

Acknowledgments. This work was funded by the AvH-, DFG-, DAAD-, KDAW- and STW-Germany. Mr. S. Arghir was also supported by the Romanian Ministry of Labor/POSDRU/6/1.5/S/19. The authors would like to thank these Organizations and also HTWG-Konstanz/Mechatronics Department/IAF for their support.

References

1. Baruch, et al.: A fuzzy-neural multi-model for nonlinear systems, identification and control. *Fuzzy Sets and Systems* 159, 2650–2667 (2008)
2. Baruch, I., Gortcheva, E., Thomas, F., Garrido, R.: A Neuro-Fuzzy model for nonlinear plants Identification. In: Proceedings of the IASTED International Conference Modeling and Simulation MS 1999, pp. 326–331 (1999)
3. Ionescu, F.: Nonlinear Mathematical Behavior and Modeling of Hydraulic Drive Systems. In: Proceedings of the 2nd World Congress of Nonlinear Analysts, Athens, Greece, vol. 30, part 3, pp. 1447–1461. Pergamon Press, Oxford (1996)

4. Ionescu, F., Vlad, C.I.: Tools of “HYPAS” for the Optimal Control of Electro-Hydraulic Drive Installations. In: 7th IFAC Symposium on Computer Aided Control Systems and Design, CACSD 1997, Gent, Belgium, April 28-30, pp. 311–316 (1996)
5. Ionescu, F., Vlad, C.I., Sugeno, C.: “HYPAS”-Fuzzy Controller Solutions for Electro-Hydraulic Drive Installations. In: Proc of 5th European Congress of Intelligent Techniques and Soft Computing, IFIP 1997, Aachen, Germany, September 8-11, pp. 1238–1242 (1997)
6. Ionescu, F., Vlad, C.I.: “HYPAS”and its Tools for the Optimal Control of Electron Drive Installations. Journal Benelux Organization for Automatic Control, 190–215 (1997)
7. Ionescu, F.: Model Generation, Simulation and Control of Hydraulic and Pneumatic Drive Systems with HYPAS. In: 6th Scandinavian International Fluid Power Conference, ISCFP 1999, Tampere, Finland, Mai 26-28, vol. II, pp. 947–961 (1999) ISBN 952-15-0181-2
8. Ionescu, F., Vlad, C.I.: Application of a Neuro-Fuzzy for Controller for Electro-Hydraulic Axis. In: Proceedings of 6th Scandinavian International Fluid Power Conference, SCFP 1999, Tampere, Finland, Mai 26-28, vol. II, pp. 1217–1224 (1999) ISBN 952-15-0181-2
9. Ionescu, F., Stefanou, D.: HYPAS– A Modular Structured Model Design, Simulation and Control Programming Environment. In: Proceedings of IASTED Intern Conf on Artificial and Computational Intelligence, ACI 2002, Tokyo, Japan, September 25-27, pp. 324–329. Acta Press (2002) ISBN: 0-88986 -358-X & ISSN: 1482-7913
10. Ionescu, F., Stefanou, D.: Intelligent and Allied Approaches to Hybrid Systems Modeling, Steinbeis edn., Berlin, Germany. Academic Publishing House, Sofia (2005) ISBN 3-938062-13-4, ISBN 954-322-107-3
11. Ionescu, F., et al.: Cell Micro and Nano Manipulations with a Hybrid Robot. In: Yih, T.C., Talpasanu, I. (eds.) Micro- and Nano Manipulations for Biomedical Applications. Artech House, Boston (2007) ISBN: 978-1-59693-254-8
12. Ionescu, F., Vlad, C.I., Arotaritei, D.: Advanced Control of an Electro-Hydraulic Axis. In: Bishop, R.H. (ed.) The Handbook of Mechatronics, pp. 33-1–33-28. CRC Press, USA (2003) ISBN: 0-8493-0066-5; Bishop, R.H. (ed.) The Handbook of Mechatronics, pp. 33-1–33-28. CRC Press, USA (2007) ISBN: 978049392573
13. Ionescu, F., Talpasanu, I., Kostadinov, K., Hradynarski, R., Arotaritei, D.: Closed Chain Mechanism of Micro and Nano Robot for Cell Manipulations. In: Proceedings of IASTED International Conference on Robotics and Applications, RA 2007, Würzburg, Germany, pp. 340–345 (August 29-31, 2007) ISBN:Paper:978-0-88986-685-0, CD:978-0-88986-686-7
14. Ionescu, F.: Modeling and Simulation in Mechatronics. In: Proc of IFAC Intern. Conf., MCPL, Sibiu, Romania, September 26-29, pp. 301–312 (2007) ISBN 978-973-739-481-1
15. Raol, J.R., Ionescu, F.: Sensor Data Fusion using H-Infinity Filters. In: KES 2001, Knowledge-Based Intelligent Informational Engineering Systems & Allied Technologies, Osaka, Japan, September 6-8, vol. 1, pp. 515–519 (2001) ISSN 0922-6389
16. Stratulat, F., Ionescu, F.: Position Control of the feed Drive with Nonlinear Actuator, Servo valve and Direct Measurement. In: Proceedings of ARA 28 Annual Congress, Bochum, Germany, September 7-12, pp. 387–394 (2004) ISBN 973-632-140-1
17. Yakubovich, V.A., Leonov, G.A., Gelig, A.K.: Stability of stationary sets in control systems with discontinuous nonlinearities. World Scientific Publishing Company, Singapore (2004)
18. Stratulat, F., Ionescu, F. (ed.): Linear Control Systems, Steinbeis-Edition (2009)
19. Vlad, C.I., Ionescu, F., Arotaritei, D., Zaharia, M.H.: Application of Control in Mechatronics (original in Romanian). BIT, Iasi (1999) ISBN 973-96414-9-0
20. MATLAB/Simulink, <http://www.mathworks.com/>
21. Ionescu, F.: Computer Aided Design of Hydraulic Drive Installations. In: Baltac, V., Davidoviciu, A. (eds.) Computer Assisted Design in Mechanical, Electrical and Electronic Engineering (Original in Romanian), pp. 149–180. Romanian Academy Edition, Bucharest, C.Z. 681.142-83 (1987)

Cluster Validity Measures Based on the Minimum Description Length Principle

Olga Georgieva¹, Katharina Tschumitschew², and Frank Klawonn^{2,3}

¹ Department of Software Engineering, Faculty of Mathematics and Informatics, Sofia University, 125 Tzarigradsko shose Blvd., 1113 Sofia, Bulgaria

² Department of Computer Science, Ostfalia University of Applied Sciences, Salzdahlumer Str. 46/48, D-38302 Wolfenbuettel, Germany

³ Bioinformatics and Statistics

Helmholtz Centre for Infection Research

Inhoffenstr. 7, D-38124 Braunschweig, Germany

Abstract. Determining the number of clusters is a crucial problem in cluster analysis. Cluster validity measures are one way to try to find the optimum number of clusters, especially for prototype-based clustering. However, no validity measure turns out to work well in all cases. In this paper, we propose an approach to determine the number of cluster based on the minimum description length principle which does not need high computational costs and is also applicable in the context of fuzzy clustering.

1 Introduction

Model selection refers to the problem of finding a suitable model that best represents a given data set. One of the main challenges in model selection is to find a model that is complex enough to fit the data, but not too complex which would lead to overfitting. A principal problem of model selection is that the complexity of a model and how well a model fits to the data are usually measured in different “units”. The model fit is often measured in terms of an error – like the mean square error as it is often used in regression – whereas model complexity usually depends on the number of parameters that the model has.

The minimum description length principle (MDL) [1] provides a possible theoretical approach to model selection. The main idea is as follows. Suppose we want to send the data from one sender to one receiver. Either we can send the data uncoded, instance by instance or we try to encode the data with the help of a model. The first method leads to a message of a very large length. With the second method we may have errors because the model might not be a perfect description of the data. However the length of the message is much shorter. According to the principle of Occam’s razor the simpler model should be chosen from two models, which fit the data equally well [2]. In that sense, MDL will choose the model, which yields the best compression of the data. The message length for data set D and model H is given by

$$L(H) + L(D|H) \quad (1)$$

where $L(H)$ is the length of the description of the model and $L(D|H)$ the length of the description of the data, encoded with the model H . Therefore the model with the minimal sum (1) would be chosen. This approach is called crude two-part MDL principle [2].

In this paper, we propose methods how to apply the MDL principle as a cluster validity measure for prototype based clustering – like k-means clustering – in order to determine the number of clusters. A partition of a given data set is considered as a model. The obtained clusters are defined by their prototypes as the most representative points of a data group [3][4]. Then the task is to assess the quality of the data partition by determining the optimal number of clusters. Several measures exist to estimate the goodness of the obtained partition according to criteria like the within-cluster distance, the partition density and other measures [5]. However, the majority of these measures do not take the complexity of the clustering model, i.e. number of clusters into account. Therefore, they are not resistant against overfitting.

The aim of this work is to propose an effective and reliable cluster validity measure including an efficient computation scheme for the evaluation of the quality of clustering results based on the MDL principle. For this propose $L(H)$ and $L(D|H)$ should be defined in a reasonable way. In this paper, we present two different approaches to define the length of the description of the model and the length of the description of the data given the model.

2 Related Work

Various approaches for cluster evaluation based on the MDL principle have been proposed. For instance, the approach presented in [6] is based on the assumption that the data in the cluster follows normal distribution – an assumption that is often too restrictive. In this case, $L(D|H)$ can be easily computed as the logarithm of the data likelihood given the distribution. In [7] Kernel MDL is proposed to estimate the optimal numbers of clusters. Likewise this approach assumed that the data is generated by Gaussian Mixture Model. The approach presented in [8] is applicable only to the labeled data. Other approaches (see for instance [9][10]) are based on the refined MDL principle and are therefore complex and require high computational costs. None of these approaches is applicable to fuzzy clustering. Detailed overview of the different approaches to determine the number of cluster can be seen for instance in [11].

3 MDL for Clustering

Let us consider a set $\{x_1, \dots, x_n\} \subset \mathbb{R}^m$ of n data points that are to be partitioned into c clusters. For x_k let u_{ik} denote the membership of the k -th data point to the i -th cluster and let d_{ik} be the distance between the k -th data point and the i -th cluster prototype ($i = 1, \dots, c$). m is the number of features describing each data point. $U = \{u_{ik}\}$ is a $c \times n$ binary partition matrix in case of classical crisp clustering. For fuzzy clustering, each u_{ik} can assume values between 0 and 1. In the simplest case, the cluster prototype is a single vector, i.e. simply the cluster centre, i.e. for the i -th cluster it is $v_i = \{v_{i1}, \dots, v_{im}\}$. The $c \times m$ matrix $V = \{v_i\}$ is then the prototype matrix.

Corresponding to the MDL principle, from the set of the candidate models $H_1, H_2 \dots$ the model H_k with the following property should be chosen:

$$L(H_k) + L(D|H_k) = \min_i \{L(H_i) + L(D|H_i)\} \quad (2)$$

Here, $L(H)$ and $L(D|H)$ should be defined in an appropriate way. The models are in our case partitions of the data into different numbers of clusters.

We consider the data as a message to be sent. Therefore, given the cluster partition we need to send the information about the cluster partition itself and the data encoded based on the given clustering. The cluster partition could be encoded by the cluster centres. The length of the description of the clustering is given by

$$L(H) = c \cdot m (1 + k), \quad \text{where } k = \max_V \{\log_2 \tilde{v}_{ij}\}. \quad (3)$$

Since the logarithm of the numbers smaller than one is negative, the following correction should be carried out: $\tilde{v}_{ij} = |v_{ij}| + 1$. Furthermore the coordinates of the cluster centres can be negative, therefore the absolute values of the coordinates and one additional sign bit for each cluster centre should be used.

A data point is assigned to the cluster to whose prototype it has the smallest distance. Therefore the data, given a certain cluster partition could be represented as the distances between the data points to the corresponding cluster centre.

$$L(D|H) = \sum_{i=1}^n \log_2 \tilde{d}_{ir_i}. \quad (4)$$

\tilde{d}_{ir_i} denotes the distance between data point x_i and the r_i for which the cluster centre is closest to x_i . The distances should be corrected in a similar way as already described above in Equation (2): $\tilde{d}_{ir_i} = d_{ir_i} + 1$. Hence, for a distance equal to zero, the code length would be zero as well. However, by encoding the data in such way, we will not be able to reconstruct the data points exactly, since we only know the overall distance of the data point to the corresponding cluster centre, but not the distance in each dimension. Therefore, it would be necessary to estimate the distances in each dimension. The accuracy through each dimension could be accounted by rewriting formula (4)

$$L(D|H) = \sum_{i=1}^n \sum_{p=1}^m \log_2 (d_{ir_i}^p + 1) \quad (5)$$

where $d_{ir_i}^p$ corresponds to the projection of the distance d_{ir_i} to the p -th dimension, $p = 1, \dots, m$.

Instead of the distances, we can also use the projections of the vectors $v_{r_i}x_i$: In this way, the data could be recovered completely without loss of information. Since in that case we will have also negative values, therefore one additional bit for the sign would be needed. However, since the number of these additional bits depends only on the amount of data, but not on the amount of cluster centres, the number of bits for the sign is constant for each clustering. Therefore, it is not important for the computation of the code length. Furthermore, for each data point we need the information to which

cluster this data point belongs. This can be solved by a certain order of the data. First the data assigned to the first cluster should be sent in the message, followed by the data assigned to the second cluster and so one.

$$L(D|H) = \sum_{i=1}^n \sum_{j=1}^m \log_2 (|x_{ij} - v_{r_{ij}}| + 1) \quad (6)$$

In case of fuzzy clustering, the fuzzy partition is governed by both the number of clusters and the values of the membership degrees. The best fuzzy partition has minimal a number of clusters with maximal membership degrees of each data point to one cluster. Therefore the code length for the data is rewritten as follows:

$$L(D|H) = \sum_{i=1}^n \sum_{j=1}^c \log_2 \tilde{\Delta}_{ij} \quad (7)$$

where $\tilde{\Delta}_{ij} = u_{ij} \cdot d_{ij} + 1$.

In order to account for the model accuracy of each dimension, the expression above is rewritten in the following form:

$$L(D|H) = \sum_{i=1}^n \sum_{j=1}^c \sum_{p=1}^m \log_2 \tilde{\Delta}_{ij}^p \quad (8)$$

where $\tilde{\Delta}_{ij}^p = u_{ij} \cdot d_{ij}^p + 1$, and d_{ij}^p correspond to the projection of the distance d_{ij} to the p -th dimension ($p = 1, \dots, m$).

The number of clusters is determined by clustering the data with different numbers of clusters and then choosing the result for which the introduced MDL formula $L(H) + L(D|H)$ based on Equation (3) including the correction $\tilde{v}_{ij} = |v_{ij}| + 1$ and Equation (5), Equation (6) or Equation (8), respectively.

4 Results

The presented approach has been implemented in Java and has been tested with artificial as well as with well known iris data set and the wine data set from the UCI Machine Learning Repository (<http://archive.ics.uci.edu/ml/>). The artificial dataset consist of 400 three-dimensional instances, i.e. $X = \{x_1, \dots, x_{400}\}$ and $x_i \in \mathbb{R}^3$. The data originate from four different multivariate normal distributions. Figure I shows the three dimensional scatter plot for the normalized data. It is obvious that the clusters overlap heavily.

For crisp clustering, the cluster centres were calculated with help of the simple k-means algorithm, obtaining different results for different numbers of clusters. In order to evaluate the clustering results, for each cluster partition, the code length corresponding to Equations (3), (4) and (6) was computed. As distance measure the Euclidean distance was used. The clustering result with the minimal code length is chosen as the best one. Furthermore, the separation index was also computed for each data set.

$$D = \min_{i=1 \dots c} \min_{j=i+1 \dots c} \frac{\min_{x \in C_i, y \in C_j} d(x, y)}{\max_{k=1 \dots c} \text{diam}_k} \quad (9)$$

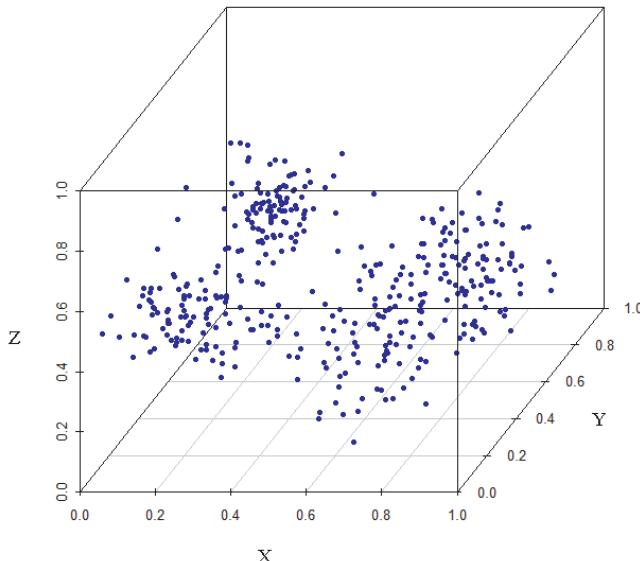


Fig. 1. Artificial dataset

In Equation (9), the numerator represents the minimal distance between clusters, which should be large. The denominator represents the diameter of the cluster C_k , which should be small for compact clusters [12]. Therefore, the separation index should be maximised.

The code length for the iris dataset for different numbers of clusters are listed in Table II. The minimum code length is indicated in bold face. Corresponding to Equation (4), two clusters should be used, whereas Equation (6) indicates three clusters as the best number of clusters. The separation index yields two as the optimal number of clusters.

Table 1. Iris dataset results

number of clusters	code length Eq. (4)	code length Eq. (6)	separation index
1	89.7	170.2	-
2	60.6	99.3	0.35811
3	61.2	90.5	0.08348
4	65.3	91.2	0.05280
5	71.3	96.7	0.05024
6	78.2	102.8	0.04275
7	83.0	106.6	0.04275
8	88.7	111.1	0.04275
9	95.2	116.8	0.05280
10	102.2	123.7	0.04759

The wine data set has 13 attributes that describe values from chemical analysis of wines grown in the same region in Italy, but coming from three different vineyards. As Table 2 shows, for the wine data set equations (4) and (6) yield one as the optimal number of clusters. However if the data should be separated in more than one cluster, three would be chosen by both equations. Which corresponds to the actual number of classes in the data. The separation index indicated eight as the optimal number of clusters.

Table 2. Wine dataset results

number of clusters	code length Eq. (4)	code length Eq. (6)	separation index
1	1477.2	1505.2	-
2	1574.5	1598.6	1.32075E-5
3	1570.8	1591.3	3.04572E-5
4	1721.9	1740.5	2.28250E-5
5	1879.3	1897.0	2.50180E-5
6	2054.1	2071.0	1.98992E-5
7	2153.3	2169.5	2.26428E-5
8	2304.7	2320.8	3.40245E-5
9	2460.4	2476.5	2.66431E-5
10	2609.1	2625.2	2.53657E-5

For the artificial data set both equations and separation index yield the same result. The optimal number of clusters according to Table 3 is four, which corresponds to the number of different distributions the data originate from.

The same calculations were made for fuzzy clustering based on the standard fuzzy c-means algorithm [3]. The code length was computed corresponding to Equations (3) and (7). As distance measure the squared Euclidean distance was used. The clustering result with the minimal code length is chosen as optimal.

Table 3. Artificial dataset results

number of clusters	code length Eq. (4)	code length Eq. (6)	separation index
1	185.4	297.5	-
2	123.5	185.3	0.07897
3	108.3	156.5	0.04165
4	98.0	137.2	0.08904
5	100.4	138.7	0.02644
6	104.2	142.0	0.02644
7	107.1	143.2	0.03217
8	111.1	146.1	0.03139
9	114.4	148.3	0.03342
10	116.5	148.6	0.03473

The results for the iris and the artificial dataset are shown in Tables 4 and 5 respectively. For the iris dataset, similar to the second column in Table 1 the clustering result with two clusters is chosen as optimal. The optimal number of clusters for the artificial dataset corresponding to the Table 5 is four, which is the correct number of clusters.

Table 4. Fuzzy clustering results: iris dataset

number of clusters	code length Eq. (7)	partition coefficient	partition entropy
1	56.34	1.00	0.00
2	41.48	0.86	0.25
3	44.27	0.74	0.47
4	51.56	0.65	0.66
5	58.09	0.59	0.79
6	65.06	0.54	0.93
7	72.76	0.50	1.04
8	79.92	0.50	1.07
9	87.08	0.48	1.14
10	94.79	0.46	1.22

Table 5. Fuzzy clustering results: artificial dataset

number of clusters	code length eq. (7)	partition coefficient	partition entropy
1	84.38	1.00	0.00
2	60.87	0.81	0.32
3	65.25	0.67	0.60
4	59.52	0.67	0.65
5	66.21	0.58	0.84
6	70.42	0.52	0.98
7	76.43	0.47	1.11
8	80.99	0.44	1.21
9	86.34	0.41	1.30
10	92.02	0.39	1.38

For comparison purposes, the validity measures partition coefficient – to be maximised – and partition entropy – to be minimised – [3] are also shown in Tables 4 and 5. Both of them cannot determine the correct number of clusters.

5 Conclusions

We have proposed a method to determine the number of clusters based on the minimum description length principle. The method works well, does not require high computational costs and can be applied in the context of ordinary as well as fuzzy clustering.

Acknowledgements. This paper was partially supported by the European Science Foundation through COST Action IC0702.

References

1. Rissanen, J.: Modeling by shortest data description. *Automatica* 14, 445–471 (1978)
2. Gruenwald, P.D.: *The Minimum Description Length Principle*. The MIT Press, Cambridge (2007)
3. Bezdek, J.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York (1981)
4. Höppner, F., Klawonn, F., Kruse, R., Runkler, T.: *Fuzzy Cluster Analysis*. John Wiley & Sons, Chichester (1999)
5. Babuska, R.: *Fuzzy Modeling for Control*. Kluwer Academic Publishers, Boston (1998)
6. Böhm, C., Goebel, S., Oswald, A., Plant, C., Plavinschi, M., Wackersreuther, B.: Integrative parameter-free clustering of data with mixed type attributes. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) *PAKDD 2010. LNCS*, vol. 6118, pp. 38–47. Springer, Heidelberg (2010)
7. Kyrgyzov, I.O., Kyrgyzov, O.O., Maître, H., Campedel, M.: Kernel mdl to determine the number of clusters. In: Perner, P. (ed.) *MLDM 2007. LNCS (LNAI)*, vol. 4571, pp. 203–217. Springer, Heidelberg (2007)
8. Banerjee, A., Langford, J.: An objective evaluation criterion for clustering. In: *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004*, pp. 515–520. ACM, New York (2004)
9. Navarro, D., Lee, M.: An application of minimum description length clustering to partitioning learning curves. In: *Proceedings of the 2005 IEEE International Symposium on Information Theory* (2005)
10. Gruenwald, P.D., Myung, I., Pitt, M.A.: *Advances in minimum description length: theory and applications*. The MIT Press, Cambridge, Massachusetts (2005)
11. Hu, X., Xu, L.: Investigation on several model selection criteria for determining the number of cluster. *Neural Inform. Proces. - Lett. and Reviews* 4 (2004)
12. Berthold, M., Borgelt, C., Höppner, F., Klawonn, F.: *Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data*. Springer, London (2010)

A Trajectory Tracking FLC Tuned with PSO for Triga Mark-II Nuclear Research Reactor

Gürçan Lokman¹, A. Fevzi Baba², and Vedat Topuz³

¹ Vocational School of Technical Science, Haliç University, Istanbul, Turkey

² Department of Electronics and Computer Education, Marmara University, Istanbul, Turkey

³ Vocational School of Technical Science, Marmara University, Istanbul, Turkey

gurcanlokman@halic.edu.tr,

{fbaba, vtopuz}@marmara.edu.tr

Abstract. In this study, a Partial Swarm Optimization tuning Trajectory Tracking Fuzzy Logic Controller (PSO-TTFLC) is designed for the nuclear research reactor Triga Mark-II in Istanbul Technical University (ITU). Reason to use TTFLC as a controller is that it uses only the error and error derivative as input parameters. TTFLC provides to work with the PSO algorithm. For this reason required changes are made in designed controller. Weights of the rules in rule base of the fuzzy controller are provided by the PSO. These parameters optimized by PSO are used to control the reactor for several working condition. Performance of the designed PSO-TTFLC is tested for various initial and desired power levels. The simulation results show that the reactor power successfully tracks the given trajectory and reaches the desired power level with the optimized weights of rules. As a result, PSO-TTFLC could control the system successfully under all conditions within the acceptable error tolerance.

Keywords: PSO, Fuzzy logic control, Trajectory tracking, Nuclear reactor control.

1 Introduction

Fuzzy logic control has been applied effectively to power control of the nuclear reactors, the best-known example being its successful application on the 5MWt Massachusetts Institute of Technology (MIT) research reactor [1]. This uses a rule-based, digital, closed-loop controller that incorporates fuzzy logic to control the power of the reactor under both steady-state and transient conditions. In addition to this, FLC is also used in controlling validated model of the pressurized water reactor (PWR)-type H.B. Robinson nuclear power plant [2], controlling water level in a simplified Advanced Boiling Water Reactor (ABWR) simulation model which was developed by GE Nuclear Energy [3], controlling the power level of the Belgium's first research reactor (BR1) [4,5].

In addition, in control of nuclear research reactors power, rod position, period and fuel temperature of the reactors were used as the controllers' input variables and they required fuzzification [6, 7]. Although they produced results superior to those of classical controllers, these approaches have difficulties for the designers to get an

insight on how the designed membership functions and fuzzy rule base affects the performance of the control system. These difficulties can be eliminated by using trajectory tracking fuzzy logic controller to represent all the variables in a single domain. A method (TTFLC) including a trajectory has been proposed to eliminate the fuzzification of such variables of the controllers by Baba in 2004 [8]. However the weights of the fuzzy rules were only determined by the user in this study. In design of controllers, if a weighted fuzzy rule base is used, the choice of any weights must be appropriate to the physical system so that simulation studies could be closer to the real system. The choice of weight values of the fuzzy rule base depends on experts' experience. Experts generally select these weight values by trial and error method. This procedure is time consuming and irksome. An expert trajectory was proposed for control of this reactor in [9]. A 2 DOF planar robot was controlled by Fuzzy Logic Controller tuned with a particle swarm optimization in [10]. To control the Triga Mark-II reactor, a Genetic Fuzzy Logic Controller was designed by Topuz and Baba [11] In our study, in order to find the optimum weights of the fuzzy rule base, a PSO algorithm was designed. Hence, the weights of the fuzzy rule base are found by PSO and insert automatically into the TTFLC. To substantiate the efficiency and effectiveness of this PSO-TTFLC, Triga Mark-II research reactor established at Istanbul Technical University in Turkey was considered.

The structure of the paper as follows: Structure of ITU Triga Mark-II research reactor in section 2. The trajectory used in this study is explained in section 3. Partial Swarm Optimization is briefly explained in section 4. Designing of a PSO-Trajectory Tracking Fuzzy Logic Control is shown in section 5. Simulation results regarding the developed control system are shown in section 6. The last section contains the conclusions.

2 Structure of ITU Triga Mark-II Research Reactor

ITU Triga1 Mark-II research reactor is an open-tank-type one with slight water coolant and graphite reflector. The reactor operates with solid fuel elements containing a homogeneous mixture of zirconium hydride moderator combined with partially enriched uranium. The reactor can be operated in two different modes named as steady-state mode and pulsed mode. It can reach a nominal power level of 250 KW at the steady-state and 1200MW at the pulsed mode in a short time. The reactor has three neutron-absorbing control rods. These are transient rod, safety rod, and regulating rod. The safety and regulating rods can be fired with an electro-mechanical mechanism. The transient rod runs with pressurized air and the electromechanical mechanism at steady-state and also with pressurized air at pulsing operation [12].

The equation representing neutronic-thermal-hydraulic behavior of reactor core is:

$$\dot{X} = AX + BU \quad (1)$$

Where: X is a representative vector of prompt neutron density, precursor concentration, fuel temperature, coolant temperature, and coolant velocity. U is a vector representing control variables. A and B are related matrixes.

In this model, mass flow rate and inlet temperature of the coolant are assumed to be time independent. However, the thermal conductivity and specific heat of the fuel and the thermal conductivity of the cladding and the physical properties of the coolant are all dependent on temperature [13].

3 The Trajectory Used in This Study

In order to define control function, it is necessary to determine the trajectory function that has to be tracked by the reactor power. The trajectory path has three elements, the first and the third elements are defined as a third order function, and the second element is defined as a second order function. The mathematical definitions of trajectory in these intervals are:

$$P_1 = a_3(t - t_0)^3 + a_2(t - t_0)^2 + a_1(t - t_0) + a_0, \quad t_0 \leq t < t_1 \quad (2)$$

$$P_2 = b_3(t - t_1)^2 + b_2(t - t_1) + b_0, \quad t_1 \leq t < t_2 \quad (3)$$

$$P_3 = c_3(t - t_2)^3 + c_2(t - t_2)^2 + c_1(t - t_2) + c_0, \quad t_2 \leq t < t_s \quad (4)$$

$$P_4 = P_s, \quad t \geq t_s \quad (5)$$

There are 14 unknown coefficients in the above equations. These coefficients can be calculated from the initial conditions, the conditions at the crossing points, the final conditions, and period conditions.

4 Partial Swarm Optimization

PSO is one of the evolutionary computation techniques proposed by Kennedy and Eberhart (1995) [14], which used group intelligence generated by cooperation and competition between group particles to guide the optimization search. Compared with other algorithms, PSO retained the global search strategy [15], used an easily operated speed-displacement model, and avoided complicated genetic operation. Its unique memory function enables it to adjust its searching strategy according to most current search situation. Thus, PSO can be more easily realized and is a more effective searching algorithm. At present, PSO have already made breakthrough progress in some practical application fields [16].

Like other evolutionary computation techniques, a population of potential solutions to the optimal problem called particles is used to probe the search-space. In PSO, every particle has a fitness value determined by object function, and is assigned a randomized velocity which determines their movement. Consider that the search space is N-dimensional space, the (i)th particle is represented by $X_i = (x_{i1}, x_{i2}, \dots, x_{iN})$ and its velocity is represented by $V_i = (V_{i1}, V_{i2}, \dots, V_{iN})$. The concrete steps of PSO are as below:

1. Random initiate particle population: In this step, the particle population (swarm) is generated randomly. This population is a space that contained many satisfaction utility preference value set, each particle represents a candidate solution of a satisfaction utility preference value set.

2. Evaluate all particles according to fitness degree value: According to fitness degree function, all satisfaction utility preference value of all particles are calculated and compared with each other.
3. Update individual extreme value according fitness degree value function: the values that derived from the previous step are ranked and the optimal solutions are found. After that, the p value of each particle is updated.
4. Update global extreme value according to fitness degree function: g is updated with optimal solution found in step 3.
5. Iterate speed and position according to update set of speed-displacement: Particle continually searches from individual extreme value p and global extreme value g to find the optimal solution.
6. Repeat step 2 to 5, until iterative condition is satisfied.

Update set of speed-displacement is:

$$V_i(k+1) = WV_i(k) + c_1 r_1 (P_i(k) - X_i(k)) + c_2 r_2 (P_g(k) - X_i(k)) \quad (6)$$

$$X_i(k+1) = X_i(k) + V_i(k+1) \quad (7)$$

$$W = W_{\max} - T \frac{W_{\max} - W_{\min}}{T_{\max}} \quad (8)$$

In these sets, position X stands for satisfaction utility preference value set, initial position X_i is the initially generated satisfaction utility preference value set; V stands for speed, P stands for individual extreme value of each particle, which means optimal satisfaction utility preference value the can be searched from the beginning iteration till present iteration; g stands for global extreme value, that means presently existed optimal satisfaction utility preference value in all solutions; c_1 and c_2 are study factors, W is inertia weight, r_1 and r_2 are random numbers in (0,1).

5 Designing of a PSO-TTFLC

The first step in designing the PSO-TTFLC is to determine the input and output variables and their ranges of TTFLC. The second step is to select the membership functions to be used in setting up the values for each input and output. The next step is to construct the weighted fuzzy control rule base. The design of fuzzy logic mainly depends on the construction of the membership functions and the fuzzy controllers. The last step is to apply the PSO algorithm to find optimum weights of rule base. The basic block diagram of the trajectory tracking fuzzy control of the research reactor is shown in Figure 1. The objective is that the reactor power follows a specified path with minimum error.

There are many forms of fuzzy reference sets for the process, and proper selection must be used to maintain high quality control of the system. The power error has five membership functions are Negative big (NB), Negative small (NS), Zero (ZE), Positive small (PS) and Positive big (PB). Where the change in power error has three membership functions are Negative (N), Zero (Z) and Positive (P). Trapezoidal

membership functions are used to represent the input fuzzy sets. The membership function is represented by membership points as shown in Figure 4. A total of Twenty-eight membership points (N_1, N_2, N_3 - Z_1, Z_2, Z_3, Z_4 - P_1, P_2, P_3 - NB_1, NB_2, NB_3 - NS_1, NS_2, NS_3, NS_4 - ZE_1, ZE_2, ZE_3, ZE_4 - PS_1, PS_2, PS_3, PS_4 - PB_1, PB_2, PB_3) is set at appropriate values by using try and error method.

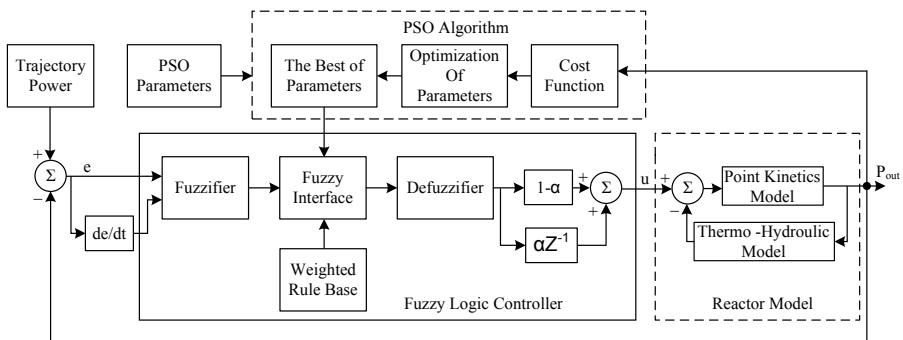


Fig. 1. Block diagram of the PSO-TTFLC

A fuzzy control action consists of situation and action pairs conditional rules based on IF and THEN statements are usually used. This study uses fifteen control rules were used in designing the PSO-TTFLC, given in Table 1.

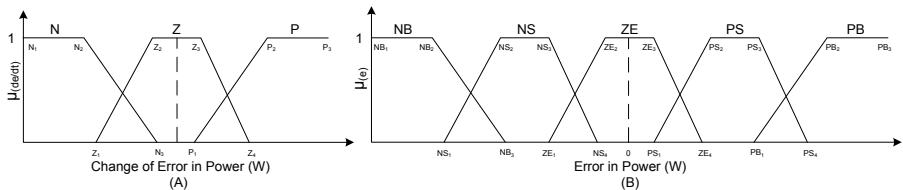


Fig. 2. (A) The membership functions of Change of Error in Power (B) The membership functions of Error in Power

The next stage involved combining the rules to obtain a final control action. The weights of fifteen rules are found by PSO and they are inserted automatically into the TTFLC. To obtain a final control action, the reactor power is measured at every sampling interval. The change in power error is calculated. The degree of membership of each variable in every labeled group is determined, and the variables can belong to more than one fuzzy set. The degree of fulfillment (DOF) of each and every rule is determined. The final control action is calculated by weighing the action of each rule by its DOF.

PSO algorithm is used to find optimum weights of rules of the TTFLC. The steps to designing a PSO-TTFLC are as follows:

1. For all particles, weights of rules are generated randomly in (-1, 1).
2. The generated weights are applied to TTFLC and the simulator is started. At the end of the simulation, fitness degree value is calculated. We use MAPE as fitness function, it can be written as;

$$MAPE = 100 \frac{1}{L} \sum_{i=1}^L \frac{|P_S - P|}{P_S} \quad (9)$$

Where the L is size of data, P_S is desired power level and P is output power.

The steps to designing a TTFLC for TRIGA Mark-II are as follows:

- The trajectory is calculated from power levels and periods
 - Reactor power is measured at every sampling interval
 - Power error and the change in power error are calculated
 - Fuzzy sets and membership functions for are determined
 - The DOF of each rule is determined.
 - The control action is determined from weighing of each action value.
3. The local best value (p) is updated according to the fitness value of each particle.
 4. The global best (g) value is updated with optimal solution in step 3
 5. Particle continually searches from p and g to find the optimal solution.
 6. Repeat step 2 to 5, until iterative condition is satisfied.
 7. Weights of rules optimized by PSO are inserted automatically into the TTFLC.

Figure 3 shows the various steps in applying PSO-TTFLC.

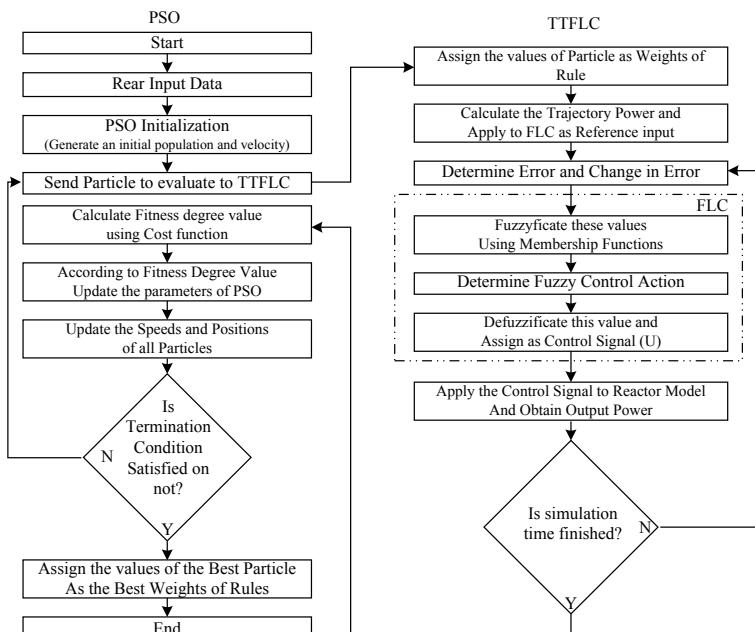


Fig. 3. Flowchart of the PSO-TTFLC for Triga Mark-II Nuclear Reactor

6 Simulation Results

In this section, simulation results regarding the developed control system are presented. PSO algorithm was started with parameters given in the table 1 and Simulator parameters are given in table 2.

Table 1. Parameters of the PSO-TTFLC

Number of Particles	40
Number of iterations	40
Learning rate	$C_1=C_2=2$
Inertia weight	Linear decreasing 0.9 to 0.4

The best MAPE value developed by PSO is 1.03048. For this system, this value is very successful. Weights found by PSO are applied to the TTFLC and the results shown in figures 4-5 are obtained. In this figure and in some others, since the error is too small, the trajectory path seems to be covered by output power.

Table 2. Parameters of Triga Mark-II Reactor Simulator

Simulation time	500 sec.
Initial Power	1kW
Desired Power	250kW
Period values	20-20-20
Sampling time	0.1 sec.

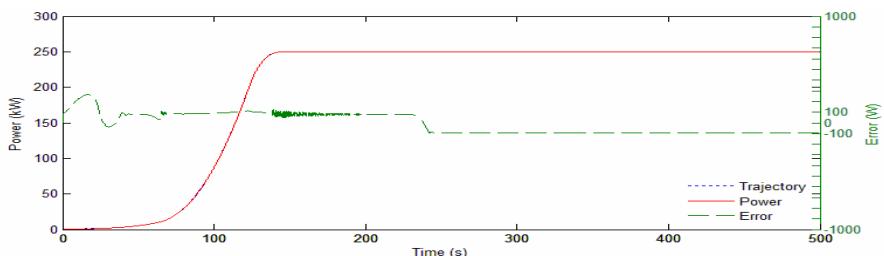


Fig. 4. Simulation result for the best values found by PSO, Trajectory, Power and Error

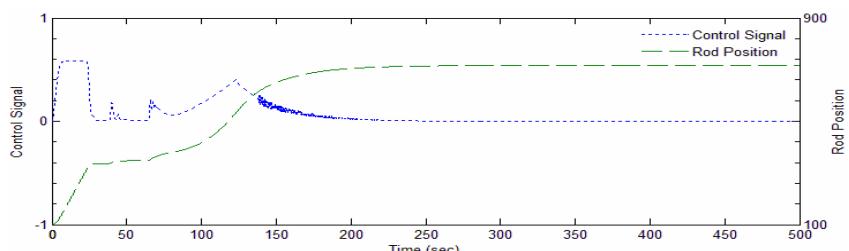


Fig. 5. Simulation result for the best values found by PSO, Control Signal and Rod Position

To test the effect of different initial power levels, the simulations has been started up for 200kW desired power level for the periods of 25-25-25 seconds and for 500W, 1kW and 5kW levels of initial power. As shown in figures 6 to 8, PSO-TTFLC is able to bring reactor power to desired level from different initial levels of power. According to initial levels, The MAPE values are 2.05, 1.05 and 2.18 respectively.

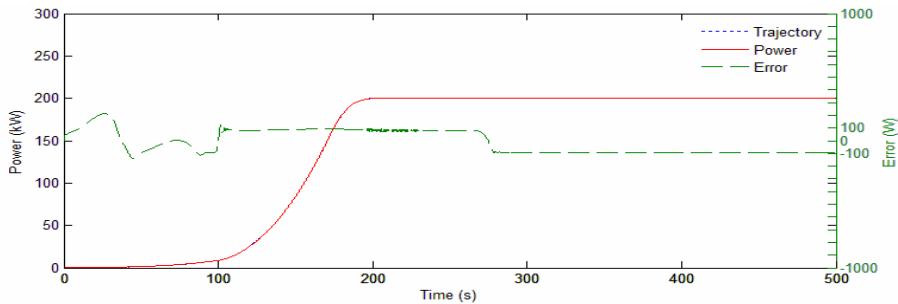


Fig. 6. $P_0=500\text{W}$, $P_S=200\text{kW}$ and period values=25-25-25

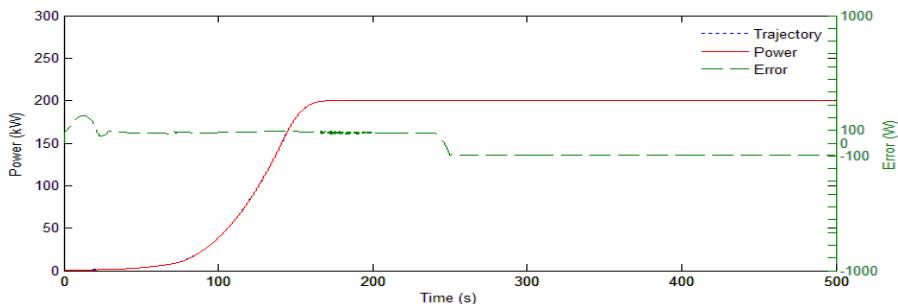


Fig. 7. $P_0=1\text{kW}$, $P_S=200\text{kW}$ and period values=25-25-25

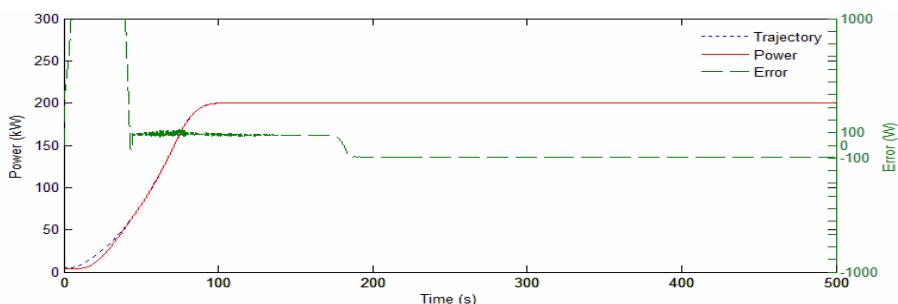


Fig. 8. $P_0=5\text{kW}$, $P_S=200\text{kW}$ and period values=25-25-25

To test the effect of different desired power levels, the simulations has been started up for 1kW initial power level for the periods of 25-25-25 seconds and for 100kW, 150kW and 250kW levels of desired power. According to desired levels, The MAPE values are 1.11, 1.05 and 1.05 respectively. Figures 9 to 11 show the results of the simulation for different desired power levels.

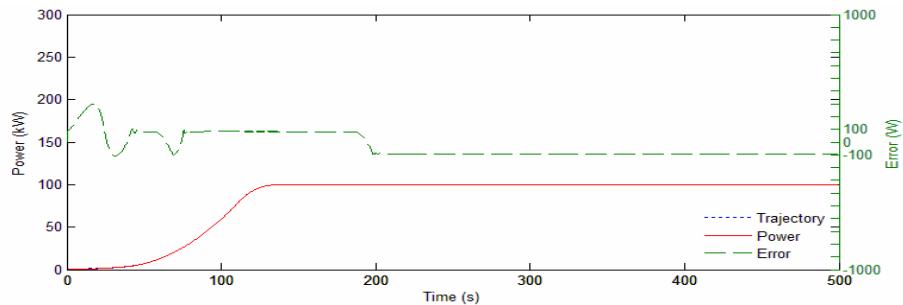


Fig. 9. $P_0=1\text{kW}$, $P_S=100\text{kW}$ and period values=25-25-25

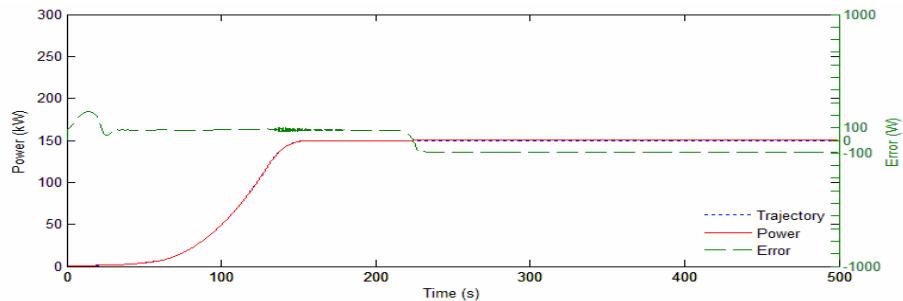


Fig. 10. $P_0=1\text{kW}$, $P_S=150\text{kW}$ and period values=25-25-25

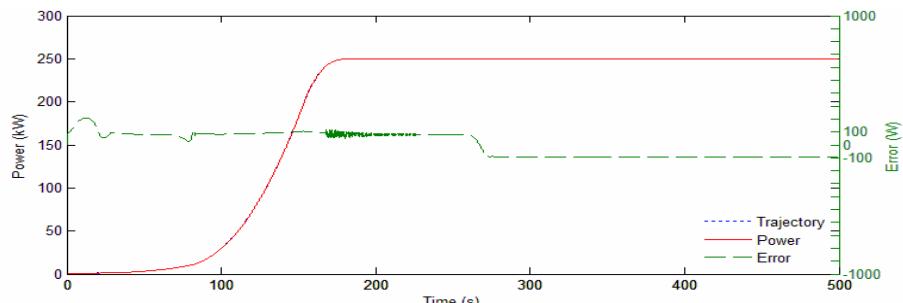


Fig. 11. $P_0=1\text{kW}$, $P_S=250\text{kW}$ and period values=25-25-25

7 Conclusions

This paper introduced the PSO based tuning method for TTFLC controller to control the Triga Mark-II research reactor established at Istanbul Technical University in Turkey. The weight values of rule base concerning the fuzzy controller were determined using PSO algorithm. In order to examine effects of the cost functions on the controller parameter optimization, MAPE was employed in PSO. Also, robustness of the controllers was tested in the case of different condition and disturbance. As a result, the PSO-TTFLC controller is able to bring reactor power to different desired levels from different initial levels of power as well as under disturbance. It is verified that the PSO-TTFLC controller has good control performance in reactor control.

References

1. Bernard, J.A.: Use of a rule-base system for process control. *IEE Control System Magazine* (1988)
2. Akin, H.L., Altin, V.: Rule-based fuzzy logic controller for a PWR-type nuclear power plant. *IEEE Trans. Nucl. Sci.* 38(2), 883–890 (1991)
3. Lin, C., Yangh, D.H.: Design of a fuzzy logic controller for water level control in an advanced boiling water reactor based on input-output data. *Nuclear Technology* 122, 318–329 (1998)
4. Ruan, D., Wal, A.: Controlling the power output of a nuclear reactor with fuzzy logic. *Information Scince*, 151–177 (1998)
5. Ruan, D.: On-Line Experiments of controlling nuclear reactor power with fuzzy logic. In: *Proceedings of IEEE International Fuzzy Systems Conference*, Seoul, Korea (August 22–25, 1999)
6. Bernard, J.A.: Use of a rule-based system for process control. *IEEE Control System Magazine* 8(5), 3–13 (1988)
7. Ruan, D., Van der Wal, A.J.: Controlling the power of a nuclear reactor with fuzzy logic. *Information Sciences* 110, 151–177 (1998)
8. Baba, A.F.: Fuzzy logic controller. *Nuclear Engineering International* 49, 36–38 (2004)
9. Coban, R., Can, B.: An expert trajectory design for control of nuclear research reactors. *Expert Systems with Applications* 36, 11502–11508 (2009)
10. Bingül, Z., Karahan, O.: A Fuzzy Logic Controller tuned with PSO for 2 DOF robot trajectory control. *Expert Systems with Applications* 38, 1017–1031 (2011)
11. Topuz, V., Baba, A.F.: Soft computing technique for power control of Triga Mark-II reactor. *Expert Systems with Applications* 38, 11201–11208 (2011)
12. Instrumentation System. Operation and maintenance manual. General Atomic Co., USA (1976); Omatsu, S., Khalid, M., Yusof, R.: *Neuro-Control and Its Applications: Advances in Industrial Control*. Springer, London (1996)
13. Can, B., Yavuz, H., Akbay, E.: The Investigation of Nonlinear Dynamics Behavior of ITU Triga Mark-II Reactor. In: *Eleventh European Triga Users Conference*, September 11-13. The German Cancer Research Institute, Heidelberg (1990)
14. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: *Proc. IEEE Int. Conf. on Neural Networks*, Piscataway, NJ, vol. IV, pp. 1942–1948 (1995)
15. Shi, Y.H., Eberhart, R.C.: A modified particle swarm optimizer. In: *IEEE International Conference on Evolutionary Computation*, pp. 69–73 (1998)
16. Kennedy, J., Eberhart, R.C., Shi, Y.: *Swarm Intelligence*. Morgan Kaufman Publishers, San Francisco (2001)

Things to Know about a (dis)similarity Measure

Lluís Belanche* and Jorge Orozco

Computer Science School
Technical University of Catalonia, Jordi Girona, 1-3
08034 Barcelona, Spain
`{belanche, jorozco}@lsi.upc.edu`

Abstract. The notions of similarity and dissimilarity are widely used in many fields of Artificial Intelligence. They have many different and often partial definitions or properties, usually restricted to one field of application and thus incompatible with other uses. This paper contributes to the design and understanding of similarity and dissimilarity measures for Artificial Intelligence. A formal dual definition for each concept is proposed, joined with a set of fundamental properties. The behavior of the properties under several transformations is studied and revealed as an important matter to bear in mind. We also develop several practical examples that work out the proposed approach.

Keywords: Similarity measures, Dissimilarity measures, Data Analysis.

1 Introduction

From a psychological point of view, a human being uses the notion of *similarity* and *dissimilarity* for problem solving, for searching information, for inductive reasoning, for element categorization, etc. For instance, the intuitive notion of similarity is to group objects under specific criteria and several theories are based on grouping objects together. This leads us to affirm that similarity and its dual concept dissimilarity are a fundamental part of many theories and applications in several fields, within or related to Artificial Intelligence, like Case Based Reasoning [1], Data Mining [2], Information Retrieval [3], Pattern Matching [4] or Neural Networks, as the Radial Basis Function network [5].

Many applications are characterized by the use of metrics for measuring differences between objects. Metric dissimilarities have been deeply studied but they are tied to a particular transitivity expression based on the triangle inequality. Very often metric (distance) functions are used due to our natural understanding of Euclidean spaces. However, not all metrics are Euclidean and many interesting dissimilarities are non-metric.

In a general sense, similarity and dissimilarity express a dual comparison between two elements. We argue that every property of a similarity should have a correspondence with one property of a dissimilarity and vice versa. This duality is commonly ignored, as well as some annoying properties (e.g. transitivity)

* Corresponding author.

and there are few general studies about how transformations of a similarity or dissimilarity can alter their properties. To worsen matters, some properties that would look natural or fundamental –like symmetry or transitivity– are still under discussion (see e.g. [6], [7], [8]). In summary, the lack of a basic agreed-upon theory sometimes leads to incompatible definitions or results focused on a specific kind of similarities or dissimilarities.

The present work intends to make a further effort in the unification of both concepts (see, for example, [9]), in two basic ways. First, with a basic but fully operational definition of similarity and dissimilarity and a set of fundamental properties and transformations. And second, with a study of how these transformations change the properties of the similarities and dissimilarities.

2 Preliminaries

Let X be a non-empty set where an equality relation is defined. In a general sense, similarity and dissimilarity express the degree of coincidence or divergence between two elements of a reference set. Therefore, it is reasonable to treat them as functions since the objective is to measure or calculate this value between any two elements of the set.

Definition 1. A similarity measure is an upper bounded, exhaustive and total function $s : X \times X \rightarrow I_s \subset \mathbb{R}$ with $|I_s| > 1$ (therefore I_s is upper bounded and $\sup I_s$ exists).

Definition 2. A dissimilarity measure is a lower bounded, exhaustive and total function $d : X \times X \rightarrow I_d \subset \mathbb{R}$ with $|I_d| > 1$ (therefore I_d is lower bounded and $\inf I_d$ exists).

Define now $s_{max} = \sup I_s$ and $d_{min} = \inf I_d$. Without loss of generality, we can take $s_{max} \geq 0$ and $d_{min} \geq 0$. In any other case, a non-negative maximum or minimum can be obtained applying a simple transformation (e.g. $s + |s_{max}|$). The following are useful properties for these functions to fulfill. For conciseness, we introduce them for both kinds of functions at the same time.

Reflexivity: $s(x, x) = s_{max}$ (implying $\sup I_s \in I_s$) and $d(x, x) = d_{min}$ (implying $\inf I_d \in I_d$).

Strong Reflexivity: $s(x, y) = s_{max} \Leftrightarrow x = y$ and $d(x, y) = d_{min} \Leftrightarrow x = y$.

Symmetry: $s(x, y) = s(y, x)$ and $d(x, y) = d(y, x)$.

Boundedness: A similarity s is lower bounded when $\exists a \in R$ such that $s(x, y) \geq a$, for all $x, y \in X$ (this is equivalent to ask that $\inf I_s$ exists). Conversely, a dissimilarity d is upper bounded when $\exists a \in R$ such that $d(x, y) \leq a$, for all $x, y \in X$ (this is equivalent to ask that $\sup I_d$ exists). Given that $|I_s| > 1$ and $|I_d| > 1$, both $\inf I_s \neq \sup I_s$ and $\inf I_d \neq \sup I_d$ hold true.

Closedness: Given a lower bounded function s , define now $s_{min} = \inf I_s$. The property asks for the existence of $x, y \in X$ such that $s(x, y) = s_{min}$ (equivalent

to asking that $\inf I_s \in I_s$). Given an upper bounded function d , define $d_{max} = \sup I_d$. The property asks for the existence of $x, y \in X$ such that $d(x, y) = d_{max}$ (equivalent to asking that $\sup I_d \in I_d$).

Complementarity: Consider now a function $C : X \rightarrow 2^X$. A lower closed similarity s defined in X has *complement function* $C(x) = \{x' \in X / s(x, x') = s_{min}\}$, if $\forall x, x' \in X, |C(x)| = |C(x')| \neq 0$. An upper closed dissimilarity d defined in X has complement function C , where $C(x) = \{x' \in X / d(x, x') = d_{max}\}$, if $\forall x, x' \in X, |C(x)| = |C(x')| \neq 0$. In case s or d are reflexive, necessarily $x \notin C(x)$. Each of the elements in $C(x)$ will be called a *complement* of x . Moreover, s or d have *unitary* complement when $\forall x \in X, |C(x)| = 1$. In this case, $\forall x \in X$:

For similarities: $\exists y' / s(x, y') = s_{max} \iff \exists y' / y' \in C(y), \forall y \in C(x)$

For dissimilarities: $\exists y' / d(x, y') = d_{min} \iff \exists y' / y' \in C(y), \forall y \in C(x)$

Let us define a *transitivity* operator in order to introduce the transitivity property in similarity and dissimilarity functions.

Definition 3. (*Transitivity operator*). Let I be a non-empty subset of \mathbb{R} , and let e be a fixed element of I . A transitivity operator is a function $\tau : I \times I \rightarrow I$ satisfying, for all $x, y, z \in I$:

1. $\tau(x, e) = x$ (null element)
2. $y \leq z \Rightarrow \tau(x, y) \leq \tau(x, z)$ (non-decreasing monotonicity)
3. $\tau(x, y) = \tau(y, x)$ (symmetry)
4. $\tau(x, \tau(y, z)) = \tau(\tau(x, y), z)$ (associativity)

There are two groups of transitivity operators: those for similarity functions, for which $e = \sup I = s_{max}$ (and then I is I_s) and those for dissimilarity functions, for which $e = \inf I = d_{min}$ (I is I_d). It should be noted that this definition reduces to uninorms [10] when $I = [0, 1]$.

Transitivity: A similarity s defined on X is called τ_s -transitive if there is a transitivity operator τ_s such that the following inequality holds:

$$s(x, y) \geq \tau_s(s(x, z), s(z, y)) \quad \forall x, y, z \in X$$

A dissimilarity d defined on X is called τ_d -transitive if there is a transitivity operator τ_d such that the following inequality holds:

$$d(x, y) \leq \tau_d(d(x, z), d(z, y)) \quad \forall x, y, z \in X$$

A similarity or dissimilarity in X may be required simply to satisfy strong reflexivity and symmetry. It is not difficult to show that strong reflexivity alone implies a basic form of transitivity \square . We call $\Sigma(X)$ the set of all similarity functions and $\Delta(X)$ the set of all dissimilarity functions defined over elements of X .

3 Equivalence

In this section we tackle the problem of obtaining *equivalent* similarities or dissimilarities, and to transform a similarity function onto a dissimilarity function or vice versa, which will naturally lead to the concept of *duality*.

3.1 Equivalence Functions

Consider the set of all ordered pairs of elements of X and denote it $X \times X$. Every $s \in \Sigma(X)$ induces a preorder relation in $X \times X$. This preorder is defined as “to belong to a class of equivalence with less or equal similarity value”. Formally, given X and $s \in \Sigma(X)$, we consider the preorder \preceq given by

$$(x, y) \preceq (x', y') \iff s(x, y) \leq s(x', y'), \forall (x, y), (x', y') \in X \times X$$

Analogously, every $d \in \Delta(X)$ induces the preorder “to belong to a class of equivalence with less or equal dissimilarity value”. Recall that $(x, y) \preceq (w, z)$ and $(w, z) \preceq (x, y)$ does *not* imply $x = w$ and $y = z$.

Definition 4. (*Equivalence*). Two similarities (or two dissimilarities) defined in the same reference set X are equivalent if they induce the same preorder.

Note that the equivalence between similarities or between dissimilarities is an equivalence relation. The properties of similarities and dissimilarities are kept under equivalence, including transitivity. The exception is the boundedness property which will depend on the chosen equivalence function. Only the *monotonically increasing* and *invertible* functions keep the induced preorder.

Definition 5. (*Equivalence function*). Let s be a similarity and d a dissimilarity. An equivalence function is a monotonically increasing and invertible function \check{f} such that $\check{f} \circ s$ is a similarity equivalent to s . Analogously, $\check{f} \circ d$ is a dissimilarity equivalent to d .

Theorem 1. Let s_1 be a transitive similarity and d_1 a transitive dissimilarity. Denote by τ_{s_1} and τ_{d_1} their respective transitivity operators. Let \check{f} be an equivalence function. Then:

1. The equivalent similarity $s_2 = \check{f} \circ s_1$ is τ_{s_2} -transitive, where

$$\tau_{s_2}(a, b) = \check{f}(\tau_{s_1}(\check{f}^{-1}(a), \check{f}^{-1}(b))) \quad \forall a, b \in I_{s_2}$$
2. The equivalent dissimilarity $d_2 = \check{f} \circ d_1$ is τ_{d_2} -transitive, where

$$\tau_{d_2}(a, b) = \check{f}(\tau_{d_1}(\check{f}^{-1}(a), \check{f}^{-1}(b))) \quad \forall a, b \in I_{d_2}$$

Proof. Consider only the similarity case, in which $\check{f} : I_{s_1} \rightarrow I_{s_2}$. Using the transitivity of s_1 we know that, for all $x, y, z \in X$, $s_1(x, y) \geq \tau_{s_1}(s_1(x, z), s_1(z, y))$.

Applying \check{f} to this inequality we get

$$(\check{f} \circ s_1)(x, y) \geq (\check{f} \circ \tau_{s_1})(s_1(x, z), s_1(z, y)).$$

Using $\check{f}^{-1} \circ s_2 = s_1$, we get

$$s_2(x, y) \geq (\check{f} \circ \tau_{s_1}) \left((\check{f}^{-1} \circ s_2)(x, z), (\check{f}^{-1} \circ s_2)(z, y) \right).$$

Defining τ_{s_2} as is defined in the Theorem we get the required transitivity expression $s_2(x, y) \geq \tau_{s_2}(s_2(x, z), s_2(z, y))$.

Therefore, any composition of an equivalence function and a similarity (or dissimilarity) function is another similarity (or dissimilarity) function, which is also equivalent.

3.2 Transformation Functions

Equivalence functions allow us to get new similarities from other similarities or new dissimilarities from other dissimilarities, but not to switch between the former and the latter. Denote by $\Sigma^*(X)$ the set of similarities defined on X with codomain on $[0,1]$ and by $\Delta^*(X)$ the set of such dissimilarities. As we shall see, using appropriate equivalence functions \check{f}^* , we have a way to get equivalent similarities (resp. dissimilarities) on $\Sigma^*(X)$ (resp. $\Delta^*(X)$) using similarities or dissimilarities on $\Sigma(X)$ (resp. $\Delta(X)$) and vice versa. In consequence, defining properties on $\Sigma(X)$ or $\Delta(X)$ is tantamount to defining them on $\Sigma(X)$ or $\Delta(X)$.

Definition 6. A $[0, 1]$ -transformation function \hat{n} is a decreasing bijection on $[0, 1]$ (implying that $\hat{n}(0) = 1$, $\hat{n}(1) = 0$, continuity and the existence of an inverse). A transformation function \hat{n} is involutive if $\hat{n}^{-1} = \hat{n}$.

This definition is restricted to (resp. dissimilarities) on $\Sigma^*(X)$ (resp. $\Delta^*(X)$). Using that both \check{f}^* and \hat{n} are bijections, a general transformation function between elements of $\Sigma(X)$ (resp. $\Delta(X)$) is the composition of two or more functions in the following way:

Definition 7. A transformation function \hat{f} is the composition of two equivalence functions and a $[0, 1]$ -transformation function:

$$\hat{f} = \check{f}_1^* \circ \hat{n} \circ \check{f}_2^{*-1},$$

where \hat{n} is a transformation function on $[0, 1]$, \check{f}_1^* obtains equivalent similarities (resp. dissimilarities) on $\Sigma(X)$ (resp. $\Delta(X)$) and \check{f}_2^* obtains equivalent similarities (resp. dissimilarities) on $\Sigma^*(X)$ (resp. $\Delta^*(X)$).

4 Duality

As it has been shown along this work, similarity and dissimilarity are two inter-related concepts. In fuzzy theory, t-norms and t-conorms are dual with respect to the fuzzy complement [12]. In the same sense, all similarity and dissimilarity functions are dual with respect to some transformation function.

Definition 8. (Duality). Consider $s \in \Sigma(X)$, $d \in \Delta(X)$ and a transformation function $\hat{f} : I_s \rightarrow I_d$. We say that s and d are dual by \hat{f} if $d = \hat{f} \circ s$ or, equivalently, if $s = \hat{f}^{-1} \circ d$. This relationship is written as a triple $\prec s, d, \hat{f} \succ$.

Theorem 2. Given a dual triple $\prec s, d, \hat{f} \succ$,

1. d is strongly reflexive if and only if s is strongly reflexive.
2. d is closed if and only if s is closed.
3. d has (unitary) complement if and only if s has (unitary) complement.
4. d is τ_d -transitive only if s is τ_s -transitive, where

$$\tau_d(x, y) = \hat{f}(\tau_s(\hat{f}^{-1}(x), \hat{f}^{-1}(y))) \quad \forall x, y \in I_d$$

Proof. Take $s \in \Sigma(X)$ and make $d = \hat{f} \circ s$.

1. For all $x, y \in X$ such that $x \neq y$, we have $s(x, y) \neq s_{\max}$; hence, applying \hat{f} , we obtain $d(x, y) \neq d_{\min}$.
2. Symmetry is immediate.
3. For all $x, y \in X$, we have $s(x, y) \geq s_{\min}$. Suppose s is closed. Since \hat{f} is strictly monotonic and decreasing, $s(x, y) > s_{\min} \Leftrightarrow (\hat{f} \circ s)(x, y) < \hat{f}(s_{\min})$. Then s is closed because there exist $x, y \in X$ such that $s(x, y) = s_{\min}$, only true if $(\hat{f} \circ s)(x, y) = \hat{f}(s_{\min})$ (i.e. if d is closed).
4. For all $x, x' \in X$ such that $x' \in C(x)$, we have $s(x, x') = s_{\min}$; applying \hat{f} , we have $(\hat{f} \circ s)(x, x') = \hat{f}(s_{\min})$; that is, $d(x, x') = d_{\max}$. Therefore, complementarity is kept.
5. For transitivity, see [12], Theorem 3.20, page 84.

Thanks to this explicit duality relation, properties on similarities are immediately translated to dissimilarities, or viceversa. A general view of all the functions and sets appeared so far is represented in Fig. 1.

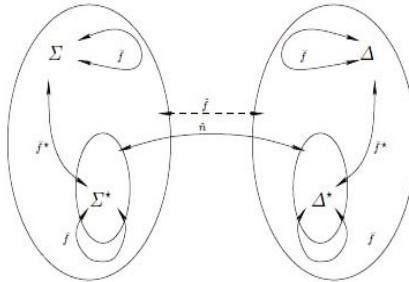


Fig. 1. Graphical representation of equivalence (\check{f}) and transformation (\hat{f}) functions from and within $\Sigma(X)$ and $\Delta(X)$

5 Application Examples

In this section we develop some simple application examples for the sake of illustration.

Example 1. Consider the dissimilarities in $\Sigma([0, 1])$ given by

$$d_1(x, y) = |x - y|, \quad d_2(x, y) = \min(x, y).$$

Their respective transitivity operators are $\tau_{d_1}(a, b) = \min(1, a + b)$ and $\tau_{d_2}(a, b) = \min(a, b)$. Consider the family of transformation functions: $\hat{f}(z) = (1 - z)^{1/\alpha}$, with $\alpha \neq 0$. The corresponding dual similarities are:

$$s_1(x, y) = (1 - |x - y|)^{1/\alpha}, \quad s_2(x, y) = \max((1 - x)^{1/\alpha}, (1 - y)^{1/\alpha}).$$

Using Theorem 2, the corresponding transitivity operators are $\tau_{s_1}(a, b) = \max(a^\alpha + b^\alpha - 1, 0)^{1/\alpha}$ and $\tau_{s_2}(a, b) = \max(a, b)$. Therefore, two dual triples are formed: $\prec s_1, d_1, \hat{f} \succ$ and $\prec s_2, d_2, \hat{f} \succ$. Note that τ_{s_1} corresponds to a well-known family of t-norms, whereas τ_{s_2} is the max norm. When $\alpha = 1$, the transitivity of s_1 is the Lukasiewicz t-norm [13].

Example 2. Consider the similarity defined in $\Sigma(\mathbb{Z})$ given by $s(x, y) = 1 - \frac{|x - y|}{|x - y| + 1}$. In this case the set I_s is the set of all rational numbers in $(0, 1]$, $\sup I_s = 1$ and $\inf I_s = 0$. This function satisfies strong reflexivity and symmetry. Moreover, it is lower bounded (with $s_{\min} = 0$), although it is not lower closed. For this reason, it does not have a complement function.

What transitivity do we have here? We know that $|x - y|$ is a metric. Consider now the transformations $\hat{n}_k(z) = z/(z + k)$, for $k > 0$. Since \hat{n}_k is subadditive, $\hat{n}_k(|x - y|)$ is also a metric dissimilarity. Therefore,

$$\frac{|x - y|}{|x - y| + 1} \leq \frac{|x - z|}{|x - z| + 1} + \frac{|z - y|}{|z - y| + 1}$$

If we apply now the transformation $\hat{n}(z) = 1 - z$, we obtain the original expression for the similarity s . Using Theorem 2, the transitivity finally changes to $s(x, y) = \max\{s(x, z) + s(z, y) - 1, 0\}$.

Example 3. Consider the function $d(x, y) = e^{|x - y|} - 1$. This is a strong reflexive and symmetric dissimilarity in $\Delta(\mathbb{R})$ with codomain $I_d = [0, +\infty)$. Therefore, it is an unbounded dissimilarity with $d_{\min} = 0$. This measure can be expressed as the composition of $\check{f}(z) = e^z - 1$ and $d'(x, y) = |x - y|$. Thus, it is τ_d -transitive with $\tau_d(a, b) = ab + a + b$. Consequently,

$$d(x, y) = d(x, z) + d(z, y) + d(x, z) \cdot (z, y), \quad \forall x, y, z \in \mathbb{R}$$

To see this, use that d' is d' -transitive with $d'(a, b) = a + b$ and Theorem 1:

$$\tau_d(a, b) = \check{f}(\check{f}^{-1}(a), \check{f}^{-1}(b)) = e^{\ln(1+a)+\ln(1+b)} - 1 = (1+a)(1+b) - 1 = ab + a + b$$

Consider now the equivalence function $\check{f} : [0, \infty) \rightarrow [0, \infty)$ given by $\check{f}(z) = \ln(z + 1)$ and apply it to the previously defined dissimilarity d . The result is the equivalent dissimilarity $d'(a, b) = |x - y|$, the standard metric in \mathbb{R} , transitive

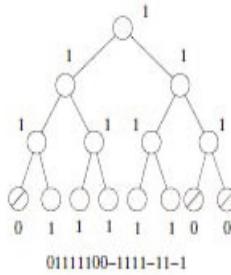


Fig. 2. A simple coding of binary trees. The reason for going bottom-up is to have the less significative digits close to the root of the tree. The choice of making the left nodes more significant than the right ones is arbitrary. The symbol \emptyset represents the empty tree.

with $d'(a, b) = a+b$ (this is the transitivity leading to the triangular inequality for metrics). The important point is that d' is also $\tau_{d'}$ -transitive, since $a+b \leq a+b+ab$ when $a, b \in [0, \infty)$. This is due to a *gradation* in the restrictiveness of transitivity operators [12]. In this case, d' is *more restrictive* than d and therefore, transitivity with the former operator implies transitivity with the latter, but not inversely.

If we apply now $f'(z) = z^2$ to d' what we get is an equivalent dissimilarity $d''(x, y) = (x-y)^2$, again strongly reflexive, symmetric and d'' -transitive, where $\tau_{d''}(a, b) = \sqrt{a^2 + b^2}$. In this case, d'' is more restrictive than both d' and d .

Similarity and dissimilarity unify preservation of transitivity using equivalence functions. This fact can be used, for example, to get a metric dissimilarity from a non-metric one. In the following example we compare the structure of two trees with a non-metric dissimilarity. Upon application of an equivalence function we get an *equivalent* and *metric* dissimilarity function.

Example 4. Consider a dissimilarity function between two binary trees. It does not measure differences between nodes but the structure of the tree. Consider a simple tree coding function D that assigns a unique value for each tree. This value is first coded as a binary number of length $2^h - 1$, being h the height of the tree. The reading of the code as a natural number is the tree code. The binary number is computed such that the most significant bit corresponds to the leftmost and bottommost tree node (Fig. 2). Note that D is not a bijection, since there are numbers that do not code a valid binary tree.

Consider now the following dissimilarity function, where A and B are binary trees. The symbol \emptyset represents the empty tree with value 0.

$$d(A, B) = \begin{cases} \max\left(\frac{D(A)}{D(B)}, \frac{D(B)}{D(A)}\right) & \text{if } A \neq \emptyset \text{ and } B \neq \emptyset \\ 1 & \text{if } A = \emptyset \text{ and } B = \emptyset \\ D(A) & \text{if } A \neq \emptyset \text{ and } B = \emptyset \\ D(B) & \text{if } A = \emptyset \text{ and } B \neq \emptyset \end{cases}$$

This is a strong reflexive, symmetric, unbounded dissimilarity with $I_d = [1, \infty)$ with $d_{min} = 1$. If we impose a limit H to the height of the trees, then d is also upper bounded and closed, $d_{max} = \sum_{i=0}^{2^H-1}$. It is also transitive with the *product* operator, which is a transitivity operator valid for dissimilarities defined on $[1, \infty)$; in other words, for any three trees A, B, C , $d(A, B) \leq d(A, C) \cdot d(C, B)$.

Proof. If neither of A, B or C are the empty tree, substituting in the previous expression and operating with max and the product we get:

$$\max\left(\frac{D(A)}{D(B)}, \frac{D(B)}{D(A)}\right) \leq \max\left(\frac{D(A)}{D(B)}, \frac{D(C)^2}{D(A)D(B)}, \frac{D(A)D(B)}{D(C)^2}, \frac{D(B)}{D(A)}\right)$$

which is trivially true. Now, if $A = \emptyset$, then the inequality reduces to $D(B) \leq \max\left(D(B), \frac{D(C)^2}{D(A)D(B)}\right)$. The cases $B = \emptyset$ or $C = \emptyset$ can be treated analogously.

If we apply now the equivalence function $\check{f}(z) = \log z$ to d we shall receive a dissimilarity $d' = \check{f} \circ d$, where the properties of d are kept in d' . However, the transitivity operator is changed using Theorem 1, to $\tau_{d'}(a, b) = a + b$. In other words, we obtain a metric dissimilarity over trees fully equivalent to the initial choice of d .

6 Conclusions

The main goal of this paper has not been to set up a standard definition of similarity and dissimilarity, but to establish some operative grounds on the definition of these widely used concepts. The data practitioner can take (or leave) the proposed properties as a guide. We have studied some fundamental transformations in order to keep these chosen basic properties. In particular, we have concentrated on transitivity and its preservation. However, a deeper study has to be done about the effects of transformations, specially in transitivity (e.g. which transformations do keep the triangle inequality) and more complex matters, like aggregation of different measures into a global one. Due to the many fields of application these concepts are involved with, the study of their properties can lead to better understanding of similarity and dissimilarity measures in many areas.

References

1. Osborne, H., Bridge, D.: Models of similarity for case-based reasoning. In: Interdisciplinary Workshop on Similarity and Categorisation, pp. 173–179 (1997)
2. Li, T., Shenghuo, Z., Ogihara, M.: A new distributed data mining model based on similarity. In: ACM SAC Data Mining Track, Florida, USA (2003)
3. Baeza-Yates, R., Ribeiro-Neto, B.: Modern information Retrieval. ACM Press, New York (1999)

4. Veltkamp, R.C., Hagedoorn, M.: Shape similarity measures, properties and constructions. In: Laurini, R. (ed.) VISUAL 2000. LNCS, vol. 1929, pp. 467–476. Springer, Heidelberg (2000)
5. Haykin, S.: Neural Networks: A Comprehensive Foundation. Prentice-Hall, Englewood Cliffs (1998)
6. Tversky, A.: Features of similarity. Psychological Review 84(4), 327–352 (1977)
7. DeCock, M., Kerre, E.: On (un)suitable relations to model approximate equality. Fuzzy Sets And Systems 133, 137–153 (2003)
8. Santini, S., Jain, R.: Similarity measures. IEEE Transactions on Pattern Analysis and Machine Intelligence (1999)
9. Bock, H., Diday, E.: Analysis of symbolic data. In: Exploratory Methods for Extracting Statistical Information from Complex Data. Springer, Heidelberg (1999)
10. Klement, E.P.: Some mathematical aspects on fuzzy sets: Triangular norms, fuzzy logics, generalized measures. Fuzzy Sets And Systems 90, 133–140 (1997)
11. Orozco, J.: Similarity and dissimilarity concepts in machine learning. Technical Report LSI-04-9-R, Universitat Politècnica de Catalunya, Barcelona, Spain (2004)
12. Klir, G., Yuan, B.: Fuzzy Sets and Fuzzy Logic: Theory and Applications. Pearson Education, London (1995)
13. Schweizer, B., Sklar, A.: Probabilistic Metric Spaces. North-Holland, Amsterdam (1983)

A New Classifier Ensembles Framework

Hamid Parvin, Behrouz Minaei-Bidgoli, and Akram Beigi

School of Computer Engineering, Iran University of Science and Technology (IUST),
Tehran, Iran
`{parvin,b_minaei,beigi}@iust.ac.ir`

Abstract. In constructing a classifier ensemble diversity is more important as the accuracy of its elements. To reach a diverse ensemble, one approach is to produce a pool of classifiers. Then we define a metric to evaluate the diversity value in a set of classifiers. We extract a subset of classifiers out of the pool in such a way that has a high diversity value. Usage of Bagging and Boosting as the sources of generators of diversity is another alternative. The third alternative is to partition classifiers and then select a classifier from each partition. Because of high similarity between classifiers of each partition, there is no need to let more than exactly one classifier from each of partition participate in the final ensemble. In this article, the performance of proposed framework is evaluated on some real datasets of UCI repository. Achieved results show effectiveness of the algorithm compare to the original bagging and boosting algorithms.

Keywords: AdaBoosting, Bagging, Classifier Ensembles, Diversity.

1 Introduction

Voting is a mechanism based on democracy form of government. This mechanism has been proven to be better than dictatorship form of government. The superiority of democracy over dictatorship is not a surprise. It is due to the fact “all are less probable to get a wrong decision”. So, ensemble methods are used in all fields inspired from the fact. Classification is a field that uses ensemble concept.

This method uses many inaccurate classifiers, instead of one accurate classifier, specialized for a few data in the different problem spaces and applies their consensus vote as the classifier. In General, it is ever-true sentence that combining diverse classifiers usually results in a better classification [5].

Diversity has a very important role in success of ensemble methods. The diversity assures the undependability of their classifiers; in the other word, the misclassifications of the classifiers don't occur simultaneously. Kuncheva [8] explains that the ensemble of a number of classifiers can always reach a better performance (even can reach a perfect accuracy) as the number of classifiers become greater, provided that they are independent (diverse).

Creating a number of classifiers diverse enough to be appropriate to participate in an ensemble is a familiar challenge. There is a very large variety of methods to reach a satisfactory diversity. Kuncheva's approach is based on the metrics that represent the amount of similarities or differences of classifier outputs.

Gianito et al. [6] imply a clustering and selection method to deal with the diversity generation. In that work, at first, a large number of classifiers with different initializations are produced, and then they select a subset of them according to their distances in their output space. They don't take into consideration how the base classifiers are created.

In this paper also a framework for development of combinational classifiers is proposed where a number of train data-bags are first bootstrapped from train data-set. Then a pool of weak base classifiers is created; each classifier is trained on one distinct data-bag. After that to get rid of similar base classifiers of the ensemble, using a clustering algorithm, here k-means, the classifiers are partitioned. The partitioning is done considering the outputs of classifiers on train dataset as feature space. In each partition, one classifier, the head of cluster, is selected to participate in final ensemble. Then, to produce consensus vote, different votes (or outputs) are gathered out of ensemble. After that the weighted majority voting algorithm is applied over them. The weights are determined using the accuracies of the base classifiers on train dataset.

The main aim of construction of An Artificial Neural Network (ANN), a model that simulates the structure and properties of biological neurons, is information processing, without necessarily creating a highly complex model of a real biological system. ANN is composed of a large number of interconnected processing elements, so called neurons, working in together to solve specific problems such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. ANN learns the input/output relationship through training with adapting weights of its connection [7].

The Multi Layer Perceptrons (MLP), the most representatives of ANNs, is a linear classifier for classifying data specified by parameters and an output function. Its parameters are adapted similar to stochastic steepest gradient descent. The units each perform a biased weighted sum of their inputs and pass this activation level through a transfer function to produce their output, and the units are arranged in a layered feedforward topology. The network thus has a simple interpretation as a form of input-output model, with the weights and thresholds (biases) the free parameters of the model. Important issues in Multilayer Perceptrons design include specification of the number of hidden layers and the number of units in these layers [7].

One way is to set the weights explicitly, using a prior knowledge. Another way is to 'train' the MLP, feeding it by teaching patterns and then letting it change its weights according to some learning rule. In this paper the MLP is used as one of the base classifiers.

Decision tree is one of the most versatile classifiers in the machine learning field. Decision tree is considered as one of the unstable classifiers that can be suitable for ensemble construction. It uses a tree-like graph or model of decisions. The kind of representation is appropriate for experts to analyze what classifier does [10]. The ensemble of a number of decision trees is a well-known ensemble called Random Forest which is one of the most powerful ensemble algorithms. The algorithm of random forest was first developed by Breiman [2]. In this paper, decision tree is totally used as one of the base classifiers.

Rest of this paper is organized as follows. Section 2 is related works. In section 3, we explain the proposed method. Section 4 demonstrates results of our proposed method against traditional ones comparatively. Finally, we conclude in section 5.

2 Related Work

Generally, there are two important challenging approaches to combine a number of classifiers that use different train sets. They are Bagging and Boosting. Both of them are considered as two methods that are sources of diversity generation.

The term Bagging is first used by [2] abbreviating for Bootstrap AGGREGATING. The idea of Bagging is simple and interesting: the ensemble is made of classifiers built on bootstrap copies of the train set. Using different train sets, the needed diversity for ensemble is obtained.

Breiman [3] proposes a variant of Bagging which it is called Random Forest. Random Forest is a general class of ensemble building methods using a decision tree as the base classifier. To be labeled a “Random Forest”, an ensemble of decision trees should be built by generating independent identically distributed random vectors and use each vector to grow a decision tree. Bagging involves having each classifier in the ensemble vote with equal weight. In order to promote model diversity, bagging trains each model in the ensemble using a randomly-drawn subset of the training set. As an example, the random forest algorithm combines random decision trees with bagging to achieve very high classification accuracy. In this paper Random Forest algorithm [8] is implemented and compared with the proposed method.

Boosting is inspired by an online learning algorithm called Hedge(β) [4]. Boosting involves incrementally building an ensemble by training each new classifier to emphasize the training instances that previous classifiers misclassified. This algorithm allocates weights to a set of strategies used to predict the outcome of a certain event. At this point we shall relate Hedge(β) to the classifier combination problem. Boosting is defined in [4] as related to the “general problem of producing a very accurate prediction rule by combining rough and moderately inaccurate rules of thumb.” The main boosting idea is to develop the classifier team D incrementally, adding one classifier at a time. The classifier that joins the ensemble at step k is trained on a dataset selectively sampled from the train dataset Z. The sampling distribution starts from uniform, and progresses towards increasing the likelihood of “difficult” data points. Thus the distribution is updated at each step, increasing the likelihood of the objects misclassified at step k-1. Here the correspondence with Hedge(β) is transposed. The classifiers in D are the trials or events, and the data points in Z are the strategies whose probability distribution we update at each step. The algorithm is called AdaBoost which comes from ADAPTIVE BOOSTING. In some cases, boosting has been shown to yield better accuracy than bagging, but it also tends to be more likely to overfit the training data. By far, the most common implementation of Boosting is AdaBoost, although some newer algorithms are reported to achieve better results. One version of these algorithms is arc-x4 which outperforms the common ADABoost [8].

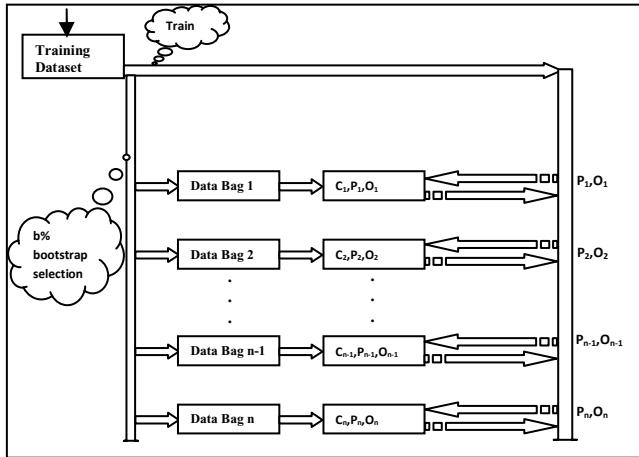


Fig. 1. Training phase of the Bagging method

3 Proposed Framework

In our proposed method, the aim is to use the most diverse set of classifiers obtained by Bagging or Boosting mechanism. In this method first, a number of classifiers are trained by the two well-known mechanisms: Bagging or Boosting and then the produced classifiers are partitioned according their outputs. Finally, the nearest classifier to the head of each produced cluster is selected.

Selection of one classifier from each cluster, and usage of them as an ensemble, can produce a diverse ensemble that outperforms the traditional Bagging and Boosting, i.e. usage of all classifiers as an ensemble, while each cluster is produced according to classifiers' outputs.

Fig. 1 illustrates the training phase of the Bagging method in general. In proposed method, it is bootstrapped n subsets of dataset with b percent of the train dataset. Then a classifier is trained on each of those subsets. In addition, it is tested each decision tree over the whole of train dataset and calculated its accuracy. O_i and P_i , denoted as i th output of classifier over train dataset and its accuracy, respectively.

Fig. 2 illustrates the training phase of the Boosting method, too. We again select a subset of dataset containing b percent of train dataset. Then the first classifier is trained on this subset. After that the first classifier is tested on the whole train dataset which this results in producing the O_1 and P_1 . Using O_1 , the next subset of b percent of train dataset is obtained. This mechanism is continued in such a way that obtaining i th subset of b percent of train dataset is produced considering the O_1, O_2, \dots, O_{i-1} . For more information about the mechanism of Boosting, the reader can refer to Kuncheva [8].

The proposed method is generally illustrated in the Fig. 3. In the proposed method we first produce a dataset whose i th dataitem is O_i . Features of this dataset are real dataitems of under-leaning dataset. Then we have a new dataset having n classifiers and N features, where n is a predefined value showing the number of classifiers produced by Bagging or Boosting and N is the cardinality of under-leaning datasets. After producing the mentioned dataset, we partition that dataset by use of a clustering

algorithm that this results in some clusters of classifiers. Each of the classifiers of a cluster has similar outputs on the train dataset; it means these classifiers have low diversities, so it is better to use one of them in the final ensemble rather than all of them. For escaping from outlier classifiers, we ignore from the clusters which contain number of classifiers smaller than a threshold.

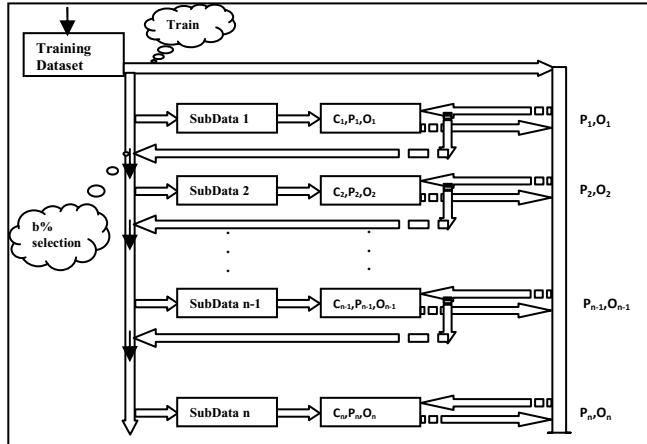


Fig. 2. Training phase of the Boosting method

Let us assume that E is the ensemble of n classifiers $\{C_1, C_2, C_3 \dots C_n\}$. Also assume that there are m classes in the case. Next, assume applying the ensemble over data sample d results in a binary D matrix like equation 1.

$$D = \begin{bmatrix} d_{1,1} & d_{1,2} & \cdot & d_{1,n} \\ \cdot & \cdot & \cdot & \cdot \\ d_{m-1,1} & d_{m-1,2} & \cdot & d_{m-1,n} \\ d_{m,1} & d_{m,2} & \cdot & d_{m,n} \end{bmatrix} \quad (1)$$

where $d_{i,j}$ is one if classifier j votes that data sample d belongs to class i . Otherwise it is equal to zero. Now the ensemble decides the data sample d to belong to class q according to equation 2.

$$q = \arg \max_{i=1}^m \left| \sum_{j=1}^n w_j * d_{i,j} \right| \quad (2)$$

where w_j is the weight of classifier j which is obtained optimally according to equation 3 [8].

$$w_j = \log \frac{p_j}{1-p_j} \quad (3)$$

where p_j is accuracy of classifier j over total train set. Note that a tie breaks randomly in equation 2.

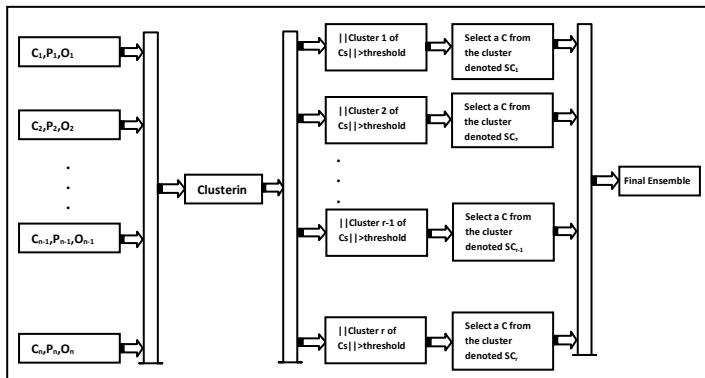


Fig. 3. Proposed method for selecting the final ensemble from a pool of classifier generated by Bagging or Boosting

Table 1. Details of used dataset

Dataset Name	# of dataitem s	# of features	# of classes	Data distribution per classes
Breast Cancer	404	9	2	444-239
Bupa	345	6	2	145-200
Glass	214	9	6	70-76-17-13-9-29
Galaxy	323	4	7	51-28-46-38-80-45-35
half-ring	400	2	2	300-100
Heart	462	9	2	160-302
Ionosphere	351	34	2	126-225
Iris	150	4	3	50-50-50
test Monk 1	412	6	2	216-216
test Monk 2	412	6	2	216-216
test Monk 3	412	6	2	216-216
train Monk 1	124	6	2	62-62
train Monk 2	169	6	2	105-64
train Monk 3	122	6	2	62-60
Wine	178	13	3	59-71-48

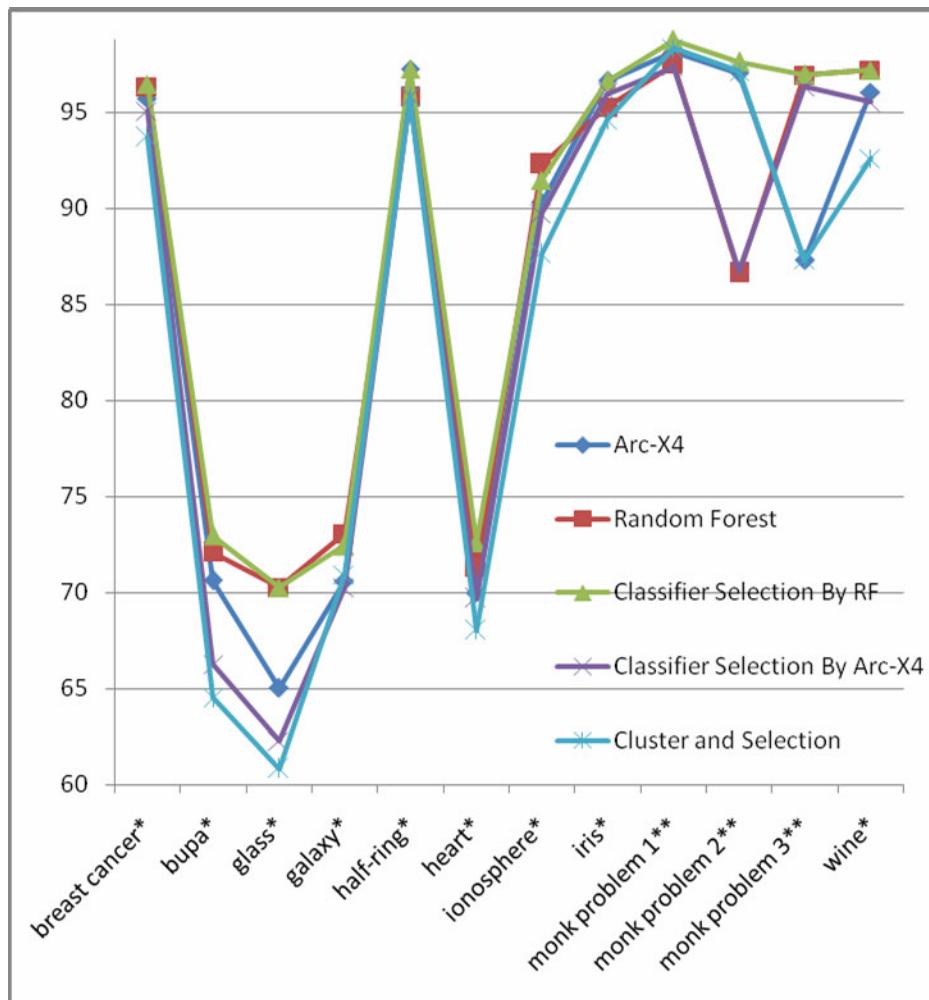


Fig. 4. Comparison of the results by considering Decision Tree as base classifier. * shows the dataset is normalized, and 4 fold cross validation is taken for performance evaluation. ** shows that the train and test sets are predefined.

4 Experimental Results

This section evaluates the result of applying the proposed algorithm on some real datasets available at USI repository [1] and one hand made dataset named half-ring. The details of half-ring dataset can be available in [9]. These dataset are summarized in the Table 1.

Table 2. Comparison of the results averaged over all 12 datasets by considering Decision Tree as base classifier

	Arc-X4	Random Forest	Classifier Selection By RF	Classifier Selection By Arc-X4	Cluster and Selection
Average	86.22	87.05	88.39	85.08	84.29

Measure of decision in each employed decision tree is taken as Gini measure. The threshold of pruning is set to 2. Also the classifiers' parameters are fixed in all of their usages. In all experiments n , r , b and threshold of accepting a cluster are set to 151, 33, 30 and 2 (i.e. only the clusters with one classifier is dropped down) respectively. All the experiments are done using 4-fold cross validation. Clustering is done by k-means with r (33) clusters.

Fig. 4 shows the accuracies of different methods over all datasets by considering a DT as each of the base classifiers. Table 2 shows the averaged accuracies Fig 4. Fig. 5 shows the accuracies of different methods over all datasets by considering a MLP as each of the base classifiers. Table 3 shows the averaged accuracies Fig. 5.

While we choose only at most 33 percent of the base classifiers of Random Forest, the accuracy of their ensemble outperforms their full ensemble, i.e. Bagging. Also it outperforms Boosting.

Because the classifiers selected in this manner (by Bagging along with clustering), have different outputs, i.e. they are as diverse as possible, they are more suitable than their all ensemble. It is worthy to mention that the Boosting is inherently diverse enough to be an ensemble totally; and the reduction of ensemble size by clustering destructs their Boosting effect. Take it in the consideration that in Boosting ensemble, each member covers the drawbacks of the previous ones.

Table 3. Comparison of the results averaged over all 12 datasets by considering MLP as base classifier

	Arc-X4	Random Forest	Classifier Selection By RF	Classifier Selection By Arc-X4	Cluster and Selection
Average	88.02	86.99	88.29	87.16	87.42

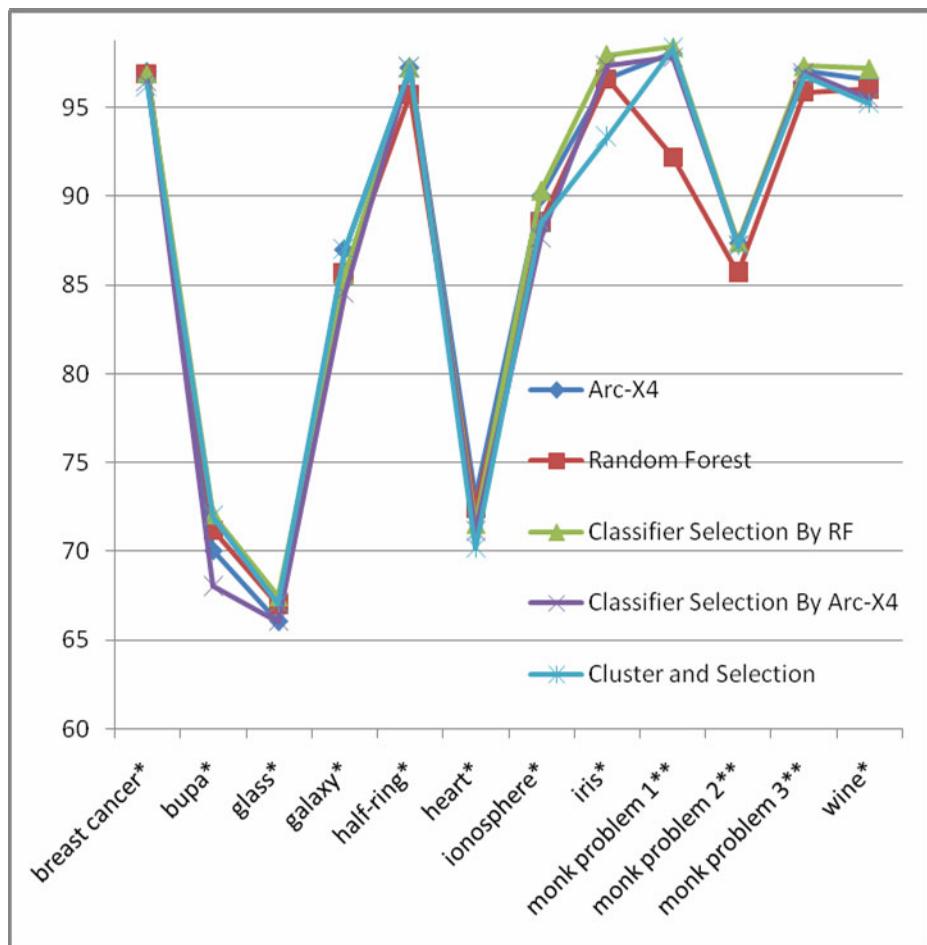


Fig. 4. Comparison of the results by considering MLP as base classifier. * shows the dataset is normalized, and 4 fold cross validation is taken for performance evaluation. ** shows that the train and test sets are predefined.

5 Conclusion and Future Work

In this paper, we have proposed a new method to improve the performance of classification. The proposed method uses Bagging as generator of the base classifiers. Then using k-means we partition the classifiers. After that we select one classifier per a validated cluster.

While we choose only at most 33 percent of the base classifiers of Bagging, the accuracy of their ensemble outperforms the full ensemble of them. Also it outperforms Boosting.

As a future work, we can turn to research on the variance of the method. Since it is said about Bagging can reduce variance and Boosting can simultaneously reduce variance and error rate.

References

1. Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases (1998),
<http://www.ics.uci.edu/~mlearn/MLRepository.html>
2. Breiman, L.: Bagging Predictors. *Journal of Machine Learning* 24(2), 123–140 (1996)
3. Breiman, L.: Random Forests. *Machine Learning* 45(1), 5–32 (2001)
4. Freund, Y., Schapire, R.E.: A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* 55(1), 119–139 (1997)
5. Giacinto, G., Roli, F.: An approach to the automatic design of multiple classifier systems. *Pattern Recognition Letters* 22, 25–33 (2001)
6. Gunter, S., Bunke, H.: Creation of classifier ensembles for handwritten word recognition using feature selection algorithms. In: IWFHR (2002)
7. Haykin, S.: Neural Networks, a comprehensive foundation, 2nd edn. Prentice Hall International, Inc., Englewood Cliffs (1999) ISBN: 0-13-908385-5
8. Kuncheva, L.I.: Combining Pattern Classifiers, Methods and Algorithms. Wiley, New York (2005)
9. Minaei-Bidgoli, B., Topchy, A.P., Punch, W.F.: Ensembles of Partitions via Data Resampling. *ITCC*, 188–192 (2004)
10. Yang, T.: Computational Verb Decision Trees. *International Journal of Computational Cognition*, 34–46 (2006)

Policy Gradient Reinforcement Learning with Environmental Dynamics and Action-Values in Policies

Seiji Ishihara¹ and Harukazu Igarashi²

¹ Kinki University, 1 Takaya-umenobe, Higashi-Hiroshima 739–2116, Japan

² Shibaura Institute of Technology, 3–7–5 Toyosu, Koto-ku, Tokyo 135–8548, Japan
ishihara@hiro.kindai.ac.jp, arashi50@sic.shibaura-it.ac.jp

Abstract. The knowledge concerning an agent’s policies consists of two types: the environmental dynamics for defining state transitions around the agent, and the behavior knowledge for solving a given task. However, these two types of information, which are usually combined into state-value or action-value functions, are learned together by conventional reinforcement learning. If they are separated and learned independently, either might be reused in other tasks or environments. In our previous work, we presented learning rules using policy gradients with an objective function, which consists of two types of parameters representing environmental dynamics and behavior knowledge, to separate the learning for each type. In such a learning framework, state-values were used as an example of the set of parameters corresponding to behavior knowledge. By the simulation results on a pursuit problem, our method properly learned hunter-agent policies and reused either bit of knowledge. In this paper, we adopt action-values as a set of parameters in the objective function instead of state-values and present learning rules for the function. Simulation results on the same pursuit problem as in our previous work show that such parameters and learning rules are also useful.

1 Introduction

Environmental dynamics is usually represented by state-transition probabilities in reinforcement learning. However, we don’t always have to know or learn environmental dynamics in advance when an agent learns its policy. For example, in Q -learning [1], which is a representative type of reinforcement learning, this information is included in action-value functions $Q(s, a)$ ($s \in S$, $a \in A$) and learned with behavior knowledge for solving a task given to an agent. Therefore, both dynamics around an agent and knowledge for solving the task are learned simultaneously in action-value functions by Q -learning. The optimal policy for an agent is calculated by a greedy search of the action-value functions. This situation pertains even in TD -learning [2], where state-value functions $V(s)$ ($s \in S$) depend on environmental dynamics as $Q(s, a)$ in Q -learning.

What defines the environmental dynamics of an agent? At least two types of factors affect it. The first is the behavior characteristics of each agent, such as

the moving characteristics of a real mobile robot. The second is such environmental conditions as a muddy field or a strong wind. As an example, consider a pursuit problem in which robot agents pursue and catch a prey robot. In such a problem, it may happen that a real robot has its own moving characteristics and the floor is very slippery. Then the environment's state transitions around the learning agents are stochastic. If we change a hunter robot to another robot with different moving characteristics or conduct a pursuit experiment in another room with a different type of floor material, we can no longer use the state-value or action-value functions obtained by previous learning experiments. If environmental dynamics and behavior knowledge are separated in an agent policy, either can be reused in other pursuit experiments where the dynamics or the task is changed. Fortunately, the former knowledge can be measured in advance or observed in a learning process. Of course, it can also be learned with the latter knowledge. Behavior knowledge independent of the environmental dynamics can be learned by simulation where only robot agents with standard deterministic moving characteristics are used. If both types of knowledge are obtained and used as initial values of parameters in an agent policy, this will greatly help reduce learning costs.

In our previous paper, we proposed a learning method to separate the learning for environmental dynamics from that for behavior knowledge [2]. Our method uses policy gradients with an objective function that consists of two types of parameters representing either environmental dynamics or behavior knowledge. In our learning framework, the decision problem of each time step of an agent is regarded as a minimization problem of an objective function, and the parameters in the function are updated to increase the expected return given to the agent. We adopted state-values as an example of the set of parameters corresponding to behavior knowledge. The simulation results on a pursuit problem showed that our method learned hunter-agent policies properly and reused either knowledge included in the policies. In this paper, we present our learning rules under the condition where action-values are used instead of the state-values. Moreover, we show the usefulness of our method by simulations on the same pursuit problem as that in the previous work.

This paper is organized as follows. Section 2 presents our learning framework based on a policy gradient method. Section 3 divides the knowledge concerning an agent's policy into environmental dynamics and behavior knowledge and formulates the learning rule for each piece of knowledge. Experimental results on a pursuit problem, discussion, and conclusion are given in Sections 4 and 5.

2 Policy Gradient and Learning Rule

2.1 Reinforcement Learning Based on Policy Gradient

Policy gradient methods originate from the REINFORCE algorithm of Williams [3]. In the REINFORCE algorithm, an agent policy includes parameters that are updated using policy gradient vectors to increase the expected return given

to an agent. REINFORCE was extended to Partially Observable Markov Decision Processes (POMDPs) by Kimura [4]. In these methods, an agent policy is learned directly without calculating state-value function $V(s)$ or action-value function $Q(s, a)$. Consequently, there was a large discrepancy between value-based reinforcement learning algorithms and primitive policy gradient methods. However, methods where policy gradient vectors are represented and calculated by action-value functions have been proposed assuming Markov Decision Processes (MDPs) [5][6]. We showed that the REINFORCE algorithm can be applied without requiring Markov properties on state transitions, rewards [7], and policies. Moreover, we derived the learning rule proposed in Refs. 5 and 6 from the learning rule of REINFORCE using the statistical properties of characteristics eligibilities [8] and showed that a policy gradient method can be applied to learning in multi-agent systems as pursuit problems [9]. We approximated the policy function controlling all agents by the product of the policy functions of each agent [10]. Therefore, the primitive policy gradient method, REINFORCE by Williams, has been extended to learning agents in POMDPs, MDPs, non-MDPs, and multi-agent systems.

2.2 Objective Function and Policy

We use a stochastic policy defined by a Boltzmann distribution function in which an objective function is used as energy. Objective function $E(a; s)$ evaluates action $a \in A$ of an agent in environmental state $s \in S$ and includes parameters that are learned to maximize the expectation of the rewards given to an agent. This paper deals with stochastic environmental dynamics as presented in Ref. 2, and we verify whether the policy gradient method with an objective function using action-values can be applied under such environments.

We define policy $\pi(a; s)$ as

$$\pi(a; s) \equiv \frac{e^{-E(a; s)/T}}{\sum_{b \in A} e^{-E(b; s)/T}}, \quad (1)$$

where T denotes temperature. When an agent is in a deterministic environment, a two-dimensional table,

$$E(a; s) = -\theta(s, a) \quad (2)$$

was used in our previous work on pursuit problems [9]. We present other objective functions for stochastic dynamics in Section 3.

2.3 Learning Rule

In this paper, we only consider episodic learning. State $s(t)$ and action $a(t)$ appear at time t . An episode consists of $\{a(t)\}$ and $\{s(t)\}$, which are time-series data on actions and states that an agent actually took and occupied in

the episode. Let μ be a parameter in objective function $E(a; s, \mu)$. The gradient vector of the expectation of reward r given in an episode can be written as [9]

$$\frac{\partial E[r]}{\partial \mu} = E \left[r \sum_{t=0}^{L-1} e_\mu(t) \right], \quad (3)$$

where L is an episode's time length from start to goal and $e_\mu(t)$ is the characteristic eligibility [3] defined by

$$e_\mu(t) = \frac{\partial}{\partial \mu} \ln \pi(a(t); s(t), \mu). \quad (4)$$

If an agent policy is given by Boltzmann-type action selection in Eq. (1) [9],

$$e_\mu(t) = -\frac{1}{T} \left[\frac{\partial E(a(t); s(t), \mu)}{\partial \mu} - \left\langle \frac{\partial E}{\partial \mu} \right\rangle_\pi \right]. \quad (5)$$

Operation $\langle \dots \rangle_\pi$ means taking an expectation weighted by probability density function $\pi(a; s, \mu)$ in Eq. (1).

We use the following learning rule based on the property of the gradient vector in Eq. (3) [8][9],

$$\Delta \mu = \varepsilon r \sum_{t=0}^{L-1} e_\mu(t), \quad (6)$$

where $\varepsilon (> 0)$ is a learning ratio and $\sum e_\mu(t)$ means the amount of eligibility that parameter μ should be reinforced by reward r . Parameter μ is updated at the end of each episode by Eq. (6).

3 Separation of Knowledge in an Agent's Policy

3.1 Objective Function in Stochastic Environments

For stochastic environments, in Ref. 2, we proposed objective function $E_V(a; s, \omega, \theta_0)$ defined by

$$E_V(a; s, \omega, \theta_0) \equiv - \sum_{s'} \omega(s, s'; a) \theta_0(s'), \quad (7)$$

where parameter $\theta_0(s)$ is task-dependent knowledge that evaluates state s by a standard environmental dynamics and parameter $\omega(s, s'; a)$ represents a stochastic property of the transition from states s to s' when an agent takes action a . An agent does not always know the accurate state-transition probabilities given by the agent's environments before learning. If an agent has the exact information or can learn it by observation while learning an agent policy, the agent can use the information as $\omega(s, s'; a)$ in Eq. (7) to appropriately select its action at each time.

Objective function $E_V(a; s, \omega, \theta_0)$ in Eq. (7) means that action a of an agent in state s should be evaluated by state-value $\theta_0(s')$ of state s' to which an agent is moved from state s by action a under the standard environmental dynamics. Then state-value $\theta_0(s')$ weighted by $\omega(s, s'; a)$ will give better evaluation of action a when the state transition is caused stochastically.

To use action-values instead of the state-values in Eq. (7), we propose objective function $E_Q(a; s, \omega, \theta_0)$ defined by

$$E_Q(a; s, \omega, \theta_0) \equiv - \sum_{s'} \omega(s, s'; a) \sum_{a'} \theta_0(s, a') P_0(a'|s, s'), \quad (8)$$

where parameter $\theta_0(s, a)$ is task-dependent knowledge that evaluates sets of state s and action a under the standard environmental dynamics in which the behavior characteristics of each agent and environmental conditions are specified by the system designer. Parameter $P_0(a'|s, s')$ represents the probability that an agent takes action a under the standard environmental dynamics when state s changes to state s' .

To consider the meaning of objective function $E_Q(a; s, \omega, \theta_0)$ in Eq. (8), suppose a situation where an agent was moved from states s to s' when it selected action a under a stochastic environmental dynamics approximated by $\omega(s, s'; a)$. In Eq. (8), the validity of the selection of the agent's action is evaluated by action-value $\theta_0(s, a')$ of action a' that moves the agent from states s to s' under standard environmental dynamics. Action a' is estimated by

$$P_0(a'|s, s') = P_0(s'|s, a') P_0(a'|s) / P_0(s'|s). \quad (9)$$

Then action-value $\theta_0(s, a)$ weighted by $P_0(a'|s, s')$, and $\omega(s, s'; a)$ will give better evaluation of action a when the state transition is caused stochastically. If state-transition probability $P_0(a'|s, s')$ of the standard environmental dynamics is deterministic and each action causes different states from state s , $P_0(a'|s)$ equals $P_0(s'|s)$. Therefore, Eq. (9) becomes:

$$P_0(a'|s, s') = P_0(s'|s, a'). \quad (10)$$

We used Eq. (10) in our simulations described in Section 4.

3.2 Characteristic Eligibilities and Learning Rules

If Eq. (8) is substituted into Eq. (5), we obtain

$$e_{\theta_0(s,a)}(t) = \frac{1}{T} \delta_{s,s(t)} \sum_{s'} P_0(a|s(t), s') [\omega(s(t), s'; a(t)) - \langle \omega(s(t), s'; a(t)) \rangle_\pi] \quad (11)$$

and

$$e_{\omega(s,s';a)}(t) = \frac{1}{T} \delta_{s,s(t)} \sum_{b'} \theta_0(s(t), b') P_0(b'|s(t), s') [\delta_{a,a(t)} - \pi(a, s(t))] , \quad (12)$$

where function $\delta_{x,y}$ takes 1 if $x = y$ and otherwise 0. Substituting the eligibilities in Eqs. (11) and (12) into Eq. (6) easily derives the following learning rules for stochastic environments:

$$\Delta\theta_0(s, a) = \frac{\varepsilon_{\theta_0} r}{T} \sum_{t=0}^{L-1} \delta_{s,s(t)} \sum_{s'} P_0(a|s(t), s') [\omega(s(t), s'; a(t)) - \langle \omega(s(t), s'; a(t)) \rangle_\pi] \quad (13)$$

and

$$\Delta\omega(s, s'; a) = \frac{\varepsilon_\omega r}{T} \sum_{t=0}^{L-1} \delta_{s,s(t)} \sum_{b'} \theta_0(s(t), b') P_0(b'|s(t), s') [\delta_{a,a(t)} - \pi(a; s(t))] \quad (14)$$

where ε_{θ_0} and ε_ω are the learning ratios that take a small positive value.

Based on Eq. (13), when state s equals state $s(t)$ and an agent took action $a(t)$, which is capable of moving the agent from states $s(t)$ to s' , action-value parameter $\theta_0(s, a)$ of a set of state s and action a is updated in proportion to the deviation of $\omega(s(t), s'; a(t))$ from its expectation value. Since the deviation in Eq. (13) is weighted by probability $P_0(a|s(t), s')$, action-value $\theta_0(s(t), a)$ is reinforced when the agent probably takes action a under the standard environmental dynamics. Note that expectation value $\langle \omega(s(t), s'; a(t)) \rangle_\pi$ does not depend on action $a(t)$ at time t , since it is defined by

$$\langle \omega(s(t), s'; a(t)) \rangle_\pi = \sum_{a'} \omega(s(t), s'; a') \pi(a'; s(t)). \quad (15)$$

Parameter $\omega(s, s'; a)$ is updated by calculating the right-hand side of Eq. (14) when state s equals state $s(t)$. When the agent probably takes action b' to move from states $s(t)$ to s' under the standard environmental dynamics, parameter $\omega(s(t), s'; a)$ is reinforced in proportion to action-value parameter $\theta_0(s(t), b')$. Parameter $\omega(s(t), s'; a)$ is increased when $a = a(t)$, because what is inside of the brackets [] in Eq. (14) does not take any negative value. On the other hand, $\omega(s(t), s'; a)$ for action a that is not $a(t)$ is decreased so that the transition from states $s(t)$ to s' caused by action a is suppressed, because what is inside of the brackets, i.e., $-\pi(a; s(t))$, takes a negative value. The degree of increasing/decreasing is reinforced in Eq. (13)/(14) by reward $r(\sigma)$ given to an agent at the end of episode σ .

4 Simulation

4.1 Pursuit Problem with Stochastic Dynamics

We conducted the same pursuit experiments as shown in Ref. 2 to verify whether dynamics parameters and action-value parameters in an agent policy can be learned by the learning rules in Eqs. (13) and (14).

Consider a 2D grid world that has a torus structure and consists of 7 by 7 squares, two hunter agents, and a prey agent. Hunters pursue the prey until all

hunters are adjacent to the square occupied by the prey. Only one agent can occupy the same square at one time. Agents move in a given order. The prey agent moves at random and does not learn anything. An episode ends when hunters catch the prey. The initial positions of the hunters and a prey are given randomly at the start of every episode. At each time step in an episode, a hunter takes and selects one action among five, including "stop" and moves from the current square that it occupies to one of the four adjacent squares. The agent's next position is determined by state-transition probabilities $P(s'|s, a)$.

4.2 Experimental Conditions

The goal of the pursuit problems described in the previous section is to catch the prey as soon as possible. To solve these problems by reinforcement learning, we give hunter agents reward r as $r = 1/L^2$ at the end of each episode. Here L is the length of an episode. As described in the next sections, we conducted learning experiments when the environmental dynamics was deterministic in Expt. 1 (Section 4.3) and stochastic in Expts. 2.1 to 2.4 (Section 4.4). In each learning experiment, we updated parameters $\omega(s, s'; a)$ and $\theta_0(s, a)$ 200,000 times and repeated each experiment ten times. The initial values of parameters $\omega(s, s'; a)$ were set to 0.2, and those of $\theta_0(s, a)$ were selected at random from $[0, 0.1]$ in each experiment. Temperature parameter T was set to 0.8. Learning ratios ε_{θ_0} and ε_ω were set to 0.3 and 0.002. After each learning experiment, we conducted an evaluation experiment for $\omega(s, s'; a)$ and $\theta_0(s, a)$, where T was set to 0.01 and pursuit simulations were repeated for 10,000 episodes.

4.3 Experiment with Deterministic Dynamics

In Expt. 1, we assumed that hunter agents could move to any adjacent square. State transition probabilities $P(s'|s, a)$ take 1 or 0. We learned the values of $\theta_0(s, a)$ when $\omega(s, s'; a)$ was set to $P(s'|s, a)$. The average episode length observed in the last episode of ten learning experiments was 6.0. We also conducted an experiment to evaluate $\theta_0(s, a)$ obtained by learning. We set $T = 0.01$ to simulate a greedy selection of action and repeated pursuit simulation 10,000 times. A greedy policy with correct value functions gives optimal policy in value-based reinforcement learning algorithms such as TD -learning and Q -learning. The average episode length obtained in the evaluation experiment was 4.2, which is smaller than 5.3 that was obtained in our previous work [2], where function E_V in Eq. (7) was applied to the same problems as that in this paper.

4.4 Experiment with Stochastic Dynamics

In this section, we assume stochastic environmental dynamics in which a hunter agent is moved with probability p 90° right of the direction in which the agent intends to move: An agent's course is stochastically diverted 90° to the right with probability p . Then state-transition probabilities $P(s'|s, a)$ take one of the values of 0, p , or $1 - p$. Under this environment, we conducted four types of experiments: Expts. 2.1 to 2.4.

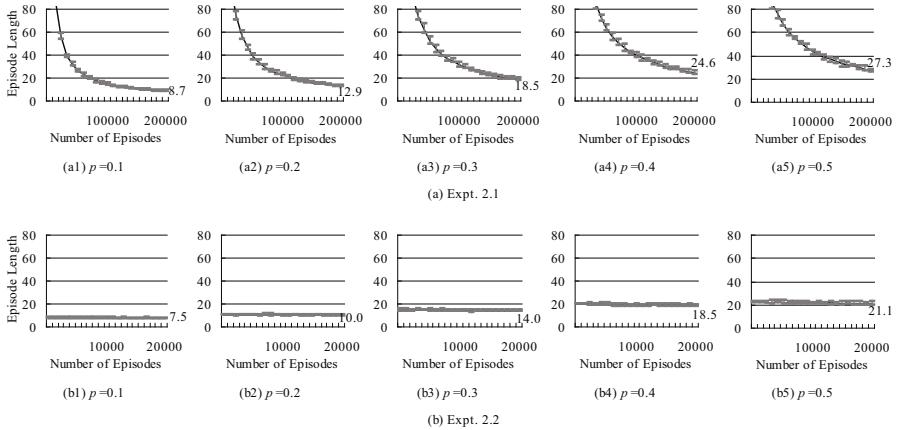


Fig. 1. Learning curves observed in Expts. 2.1 and 2.2

Table 1. Average length over 10,000 episodes generated by policy with parameters $\{\theta_0(s, a)\}$ determined from Expts. 2.1 and 2.2

	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$	$p = 0.5$
(a) Expt. 2.1	5.1	6.1	7.5	8.9	9.7
(b) Expt. 2.2	5.0	6.2	7.5	8.7	9.2

In Expts. 2.1 and 2.2, we learn $\theta_0(s, a)$ of the hunter agents when $\omega(s, s'; a)$ is set to $P(s'|s, a)$. The initial values of $\theta_0(s, a)$ are selected at random from $[0, 1]$ in Expt. 2.1. In Expt. 2.2, one set of action-value parameters $\{\theta_0(s, a)|s \in S, a \in A\}$ is selected in order from ten sets of $\{\theta_0(s, a)|s \in S, a \in A\}$ obtained when deterministic dynamics is assumed in Expt. 1. The set of action-values is used as the initial values of $\theta_0(s, a)$ in Expt. 2.2. We finished each learning experiment in Expt. 2.2 when $\theta_0(s, a)$ was updated 20,000 times because it seemed that Expt. 2.2 was easier than Expt. 2.1. Experiment 2.3 learns $\omega(s, s'; a)$ with $\theta_0(s, a)$ fixed to the set of action-values. Experiment 2.4 simultaneously learns both $\theta_0(s, a)$ and $\omega(s, s'; a)$. In Expt. 2.4, we used deterministic dynamics for the initial values of $\omega(s, s'; a)$ to accelerate the learning.

4.5 Experimental Results

The changes in episode length L averaged over 10,000 episodes are shown in Fig. 1. Figs. 1(a) and 1(b) are the learning curves obtained in Expts. 2.1 and 2.2. In Fig. 1, the learning curves are averaged over ten trials for each experiment, and the minimum and maximum among the ten trials are depicted by error bars. The average lengths over 10,000 episodes in ten evaluation experiments ($T = 0.01$) for each value of p are shown in Table 1.

Table 2. Episode length and mean square difference between $\omega(s, s'; a)$ and $P(s'|s, a)$ observed in Expt. 2.3

	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$	$p = 0.5$
Mean square difference between ω and P	0.006	0.006	0.007	0.006	0.004
Length of last episode at learning stage	8.6	10.8	14.20	19.1	24.6
Average length at evaluation stage ($T = 0.01$)	5.0	6.2	7.8	9.8	10.4

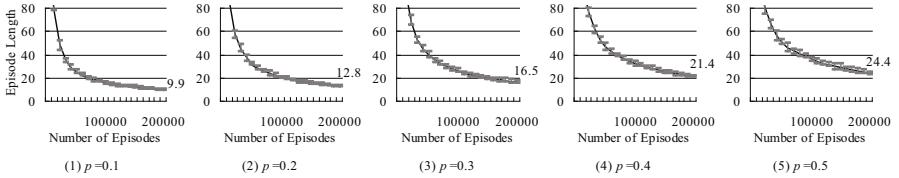


Fig. 2. Learning curves observed in Expt. 2.4

Table 3. Episode length and mean square difference between $\omega(s, s'; a)$ and $P(s'|s, a)$ observed in Expt. 2.4

	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$	$p = 0.5$
Mean square difference between ω and P	0.006	0.006	0.009	0.017	0.024
Average length at evaluation stage ($T = 0.01$)	5.1	6.2	7.4	9.0	10.0

Fig. II(a) and Table II(a) suggest that action-values $\theta_0(s, a)$ in Eq. (8) can be learned by a policy gradient method described in Section 3 if correct dynamics $P(s'|s, a)$ is given to $\omega(s, s'; a)$ in Eq. (8). From Fig. II(b) and Table II(b), we conclude that action-values $\theta_0(s, a)$ obtained by learning under environments with deterministic dynamics can be used as initial values of $\theta_0(s, a)$ even when environmental dynamics is stochastic. Such reuse as initial values greatly reduces the computation learning time.

The mean square averages between $\omega(s, s'; a)$ (i.e., the average of ten sets of $\omega(s, s'; a)$) obtained in Expt. 2.3 and the actual stochastic dynamics $P(s'|s, a)$ are shown in Table 2 for different values of p . The table also lists the average lengths of the last episode observed in ten learning experiments and over 10,000 episodes in ten evaluation experiments ($T = 0.01$) for each value of p . The results of the average lengths in the evaluation experiments suggest that an agent dynamics represented by $\omega(s, s'; a)$ in Eq. (8) can be learned if action-value parameters $\theta_0(s, a)$ are given properly in Eq. (8).

The learning curves observed in Expt. 2.4 are shown in Fig. 2. The mean square averages between $\omega(s, s'; a)$ and $P(s'|s, a)$ and the average lengths over 10,000 episodes in ten evaluation experiments ($T = 0.01$) are listed in Table 3 for each value of p . The results of the evaluation experiments shown in Table 3 indicate that simultaneous learning of all parameters in Eq. (8) can be obtained by

a policy gradient method described in Section 3 if the initial values of $\omega(s, s'; a)$ are selected properly. However, in Expt. 2.4, the mean square averages between $\omega(s, s'; a)$ and $P(s'|s, a)$ increased as probability p became larger from 0.1 to 0.5.

In all this section's experiments, the average episode lengths obtained in the evaluation experiments tended to be smaller than those in Ref. 2. Function E_Q obtained better policies than function E_V because the parameter space of the action-values has a higher dimension than the state-values.

5 Conclusion

In this paper, we proposed a separation of environmental dynamics and behavior knowledge represented by stochastic properties and action-values. Both types of parameters are included in objective function $E_Q(a; s, \omega, \theta_0)$, which is used as the policy function of an agent at each time step in policy gradient methods. We also derived the learning rules for the parameters. Objective function $E_Q(a; s, \omega, \theta_0)$ consists of the product of dynamics parameter $\omega(s, s'; a)$ and action-value parameter $\theta_0(s, a)$, which corresponds to action-value function $Q(s, a)$, weighted by state-transition probability $P_0(s'|s, a)$. Therefore, our method can separate and use knowledge corresponding to action-value function $Q(s, a)$. Such separation allows the reuse of action-value parameters for agents in different environmental dynamics. Conversely, an agent's dynamics parameters can be reused in different tasks. Moreover, when we measured the environmental dynamics for each agent in advance and obtained action-values by simulation with standard deterministic dynamics and used both values as initial parameter values in an agent's objective function, we considerably reduced the learning costs in both types of parameters, even when the dynamics is stochastic.

We considered pursuit problems where two hunter agents pursue a prey agent that moves randomly in a 7 by 7 grid world. Our experiments show that the dynamics parameters and action-value parameters in each agent policy function can be learned even if agents cannot move deterministically but are diverted 90° to the right stochastically. In the future, we will investigate our method using real mobile robots for pursuit problems in a real-world environment.

References

1. Sutton, R.S., Barto, A.G.: Reinforcement Learning. MIT Press, Cambridge (1998)
2. Ishihara, S., Igarashi, H.: Behavior Learning Based on a Policy Gradient Method: Separation of Environmental Dynamics and State Values in Policies. In: Ho, T.-B., Zhou, Z.-H. (eds.) PRICAI 2008. LNCS (LNAI), vol. 5351, pp. 164–174. Springer, Heidelberg (2008)
3. Williams, R.J.: Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. Machine Learning 8, 229–256 (1992)
4. Kimura, H., Yamamura, M., Kobayashi, S.: Reinforcement Learning by Stochastic Hill Climbing on Discounted Reward. In: Proc. of ICML 1995, pp. 295–303 (1995)
5. Sutton, R.S., McAllester, D., Singh, S., Mansour, Y.: Policy Gradient Methods for Reinforcement Learning with Function Approximation. In: Advances in NIPS, vol. 12, pp. 1057–1063. MIT Press, Cambridge (2000)

6. Konda, V.R., Tsitsiklis, J.N.: Actor-Critic Algorithms. In: Advances in NIPS, vol. 12, pp. 1008–1014. MIT Press, Cambridge (2000)
7. Baird, L., Moore, A.: Gradient Descent for General Reinforcement Learning. In: Advances in NIPS, vol. 11, pp. 968–974. MIT Press, Cambridge (1999)
8. Igarashi, H., Ishihara, S., Kimura, M.: Reinforcement Learning in Non-Markov Decision Processes —Statistical Properties of Characteristic Eligibility—. Natural Sciences and Engineering 52(2), 1–7 (2008)
9. Ishihara, S., Igarashi, H.: Applying the Policy Gradient Method to Behavior Learning in Multi-agent Systems: The Pursuit Problem. Systems and Computers in Japan 37(10), 101–109 (2006)
10. Peshkin, L., Kim, K.E., Meuleau, N., Kaelbling, L.P.: Learning to cooperative via policy search. In: Proc. of UAI 2000, pp. 489–496 (2000)

Fuzzy c -Means Clustering with Mutual Relation Constraints

Construction of Two Types of Algorithms

Yasunori Endo¹ and Yukihiro Hamasuna²

¹ Department of Risk Engineering,

Faculty of Systems and Information Engineering, University of Tsukuba
1-1-1, Tennodai, Tsukuba, Ibaraki 305-8573, Japan

endo@risk.tsukuba.ac.jp

² Department of Informatics,

School of Science and Engineering, Kinki University
3-4-1, Kowakae, Higashiosaka, Osaka 577-8502, Japan

yhama@info.kindai.ac.jp

Abstract. Recently, semi-supervised clustering attracts many researchers' interest. In particular, constraint-based semi-supervised clustering is focused and the constraints of must-link and cannot-link play very important role in the clustering. There are many kinds of relations as well as must-link or cannot-link and one of the most typical relations is the trade-off relation. Thus, in this paper we formulate the trade-off relation and propose a new "semi-supervised" concept called mutual relation. Moreover, we construct two types of new clustering algorithms with the mutual relation constraints based on the well-known and useful fuzzy c -means, called fuzzy c -means with the mutual relation constraints.

1 Introduction

Clustering is known as a very useful tool in many fields for data mining and we can find the structure of datasets through the clustering methods.

Now, semi-supervised clustering attracts many researchers' interest, e.g., Refs. [1,2,3,4,5]. The semi-supervised clustering uses a small amount of labeled data to aid and bias the clustering of unlabeled data. The clustering can be divided broadly into two categories, one is constraint-based and the other is distance-based. In the former, the objective functions of the clustering are modified for including the given constraints and then, the constraints are enforced during clustering process. In the latter, a distance function are trained on the supervised dataset to satisfy the given labels or constraints and then, it is applied to the complete dataset.

In particular, the constraints proposed by Wagstaff et al. in Ref. [2] are very interesting. They constructed a constrained clustering which is a class of semi-supervised clustering and two constraints of must-link and cannot-link play very important role in the clustering. The constraints are given to some pairs of data

and those mean prior knowledge whether the pair should be classified into one cluster or not. Concretely, the must-link constraint specifies that two data have to be placed in the same cluster, and the cannot-link constraint specifies that two data must not be placed in the same cluster.

By the way, there are many kinds of relations as well as must-link or cannot-link. One of the most typical relations is the trade-off relation. Thus, in this paper we consider the trade-off relation and formulate the relation as one of constraints called mutual relation constraint. To formulate the trade-off relation, we introduce vectors called mutual relation vectors. The range of each mutual relation vector is provided in advance and the mutual relation vector is calculated based on its range, the distance between data and so on. Each datum is represented as the sum of a given vector and the mutual relation vector. That is, it can not be determined whether a group of data in the trade-off relation are placed in the same cluster or not in advance.

In this paper, we construct two types of new clustering algorithms with the mutual relation constraints based on the well-known and useful fuzzy c -means (FCM), called fuzzy c -means with the mutual relation constraints (FCMMR).

2 Mutual Relation

2.1 Preliminaries

First of all, we define some symbols.

Each data is denoted $x_k = (x_{k1}, \dots, x_{kp})^T \in \mathbb{R}^p$ and the dataset $X = \{x_1, \dots, x_n\}$ is given. Each cluster $C_i (i = 1, \dots, c)$ has a cluster center $v_i = (v_{i1}, \dots, v_{ip})^T \in \mathbb{R}^p$. V means a set of cluster centers $\{v_1, \dots, v_c\}$. A membership grade for x_k to C_i which means belongingness of x_k to C_i is denoted by u_{ki} . U means a partition matrix $(u_{ki})_{1 \leq k \leq n, 1 \leq i \leq c}$. The result of the clustering is obtained as U .

2.2 Mutual Relation

As mentioned in Section 1, Wagstaff et al. proposed constrained clustering in Ref. [2] which is a class of semi-supervised clustering with two constraints of must-link and cannot-link.

By the way, there are many kinds of relations as well as must-link or cannot-link. One of the most typical relations is the trade-off relation. Now, let us consider the following example of the relation.

A buyer of cellphone, of course we can give other examples of car or PC, first imagines a pattern space of cellphones with considering some of their attributes, e.g., the functions or prices, and he maps those into the pattern space before deciding the purchase. Next, he try to classify those into two, three or four clusters. The number of clusters depends on their attributes of cellphones. In case of cellphones, the number of clusters is often two or three. On the other hand, makers of cellphones develop some cellphones with considering which cluster each cellphone belongs to, but their resources for the development are limited

thus it is a very important problem for the makers to determine the amount of resources allocated to each cellphone. Now, let us assume that a maker tries to make two cellphones and wants to assign them to separate clusters. He would first develop each prototype and next consider the assignment. If the development is wrong, he can't assign the cellphone to the cluster as he wants no matter how he allocate the resources after developing the prototypes. Additionally, even if the development was not wrong, he also would not assign the cellphone as he wants when he has few resources.

Here, let us define x_1 and x_2 as the two prototypes and K as the total resources. Each cellphone can be represented as $x_1 + \varepsilon_1$ and $x_2 + \varepsilon_2$ using ε_1 and ε_2 under constraints for resource allocation $\|\varepsilon_1\| + \|\varepsilon_2\| \leq K$ or $\|\varepsilon_1\|^2 + \|\varepsilon_2\|^2 \leq K^2$.

Note that the maker does not determine whether $x_1 + \varepsilon_1$ and $x_2 + \varepsilon_2$ belong to the same cluster or not, but the clustering process do that because there are many cellphones developed by other makers in the market. That is, the constraint differs from must-link or cannot-link in that it is not determined whether a group of data in one mutual relation are placed in the same cluster or not in advance.

Now, to formulate the above problem, we introduce a new concept of mutual relation which means trade-off relation as follows:

$$M_h = \{x_k \mid \text{All the } x_k \text{ are in one mutual relation.}\}, \quad (1)$$

$$M_h \cap M_{h'} = \emptyset, \quad \forall h, h'. \quad (h \neq h') \quad (1 \leq h, h' \leq r) \quad (2)$$

In the above example, r means the number of makers and each M_h means a group of cellphones which are developed by the h -th maker.

In this paper, we consider new clustering algorithms with the above mutual relation constraints.

3 Fuzzy c -Means Clustering with Mutual Relation Constraints

In this section, we propose two types of fuzzy c -means clustering with mutual relation constraints. One is based on standard FCM (sFCM) developed by Dunn in Ref. [6] and improved by Bezdek in Ref. [7], and the other is based on entropy-based FCM (eFCM) proposed by Miyamoto et. al in Ref. [8].

3.1 Standard FCMMR

In Refs. [6] and [7], Dunn and Bezdeck fuzzified hard c -means by MacQueen in Ref. [9] by introducing a fuzzification parameter m . The well-known and useful algorithm is called standard fuzzy c -means (sFCM). Here, we consider a fuzzy c -means clustering algorithm under the above proposed constraints based on sFCM, that is, standard fuzzy c -means clustering with mutual relation constraints (sFCMMR).

Now, we formulate sFCMMR as follows:

$$J_{\text{sFCMMR}}(U, V, E) = \sum_{k=1}^n \sum_{i=1}^c (u_{ki})^m d_{ki}, \quad (3)$$

$$\sum_{i=1}^c u_{ki} = 1, \quad \forall k, \quad (4)$$

$$\sum_{x_k \in M_h} \|\varepsilon_k\|^2 \leq (K_h)^2, \quad (K_h > 0) \quad (5)$$

$$\sum_{x_k \notin M_h} \|\varepsilon_k\|^2 = 0, \quad \forall h, \quad (1 \leq h \leq r) \quad (6)$$

where $d_{ki} = \|x_k + \varepsilon_k - v_i\|^2$. $\varepsilon_k = (\varepsilon_{k1}, \dots, \varepsilon_{kp})^T \in \Re^p$ under the constraints (5) and (6) is a vector providing an excursion of x_k . We call ε_k mutual relation vector and let E be $\{\varepsilon_k\}$.

We use the Lagrange multiplier method to find the optimal solutions. First, we introduce the function L_{sFCMMR} as follows:

$$\begin{aligned} L_{\text{sFCMMR}}(U, V, E) &= J_{\text{sFCMMR}}(U, V, E) + \sum_{k=1}^n \zeta_k \left(\sum_{i=1}^c u_{ki} - 1 \right) \\ &\quad + \sum_{h=1}^r \eta_h \left(\sum_{x_k \in M_h} \|\varepsilon_k\|^2 - (K_h)^2 \right). \end{aligned}$$

From Karush-Kuhn-Tucker conditions, the necessary conditions that the solutions should satisfy are as follows:

$$\begin{cases} \frac{\partial L_{\text{sFCMMR}}}{\partial v_i} = 0, & \frac{\partial L_{\text{sFCMMR}}}{\partial u_{ki}} = 0, & \frac{\partial L_{\text{sFCMMR}}}{\partial \varepsilon_k} = 0, \\ \frac{\partial L_{\text{sFCMMR}}}{\partial \zeta_k} = 0, \\ \frac{\partial L_{\text{sFCMMR}}}{\partial \eta_h} \leq 0, & \eta_h \frac{\partial L_{\text{sFCMMR}}}{\partial \eta_h} = 0, & \eta_h \geq 0. \end{cases}$$

For v_i ,

$$\frac{\partial L_{\text{sFCMMR}}}{\partial v_i} = -2 \sum_{k=1}^n (u_{ki})^m (x_k + \varepsilon_k - v_i) = 0.$$

Hence,

$$v_i = \frac{\sum_{k=1}^n (u_{ki})^m (x_k + \varepsilon_k)}{\sum_{k=1}^n (u_{ki})^m}.$$

For u_{ki} ,

$$\frac{\partial L_1}{\partial u_{ki}} = m(u_{ki})^{m-1} d_{ki} + \zeta_k = 0,$$

$$u_{ki} = \left(-\frac{\zeta_k}{md_{ki}} \right)^{\frac{1}{m-1}}.$$

From the constraint for u_{ki} ,

$$\sum_{i=1}^c u_{ki} = \sum_{i=1}^c \left(-\frac{\zeta_k}{md_{ki}} \right)^{\frac{1}{m-1}} = 1,$$

$$\left(-\frac{\zeta_k}{m} \right)^{\frac{1}{m-1}} = \frac{1}{\sum_{i=1}^n (1/d_{ki})^{\frac{1}{m-1}}}.$$

Hence,

$$u_{ki} = \frac{(1/d_{ki})^{\frac{1}{m-1}}}{\sum_{i=1}^c (1/d_{ki})^{\frac{1}{m-1}}}.$$

For ε_k , we have to consider two cases, $x_k \in M_h$ and $x_k \notin M_h$. In case that $x_k \notin M_h$, from (6),

$$\varepsilon_k = 0. \quad (x_k \notin M_h, \forall h)$$

In case that $x_k \in M_h$,

$$\frac{\partial L_{\text{sFCMMR}}}{\partial \varepsilon_k} = \sum_{i=1}^c 2(u_{ki})^m (x_k + \varepsilon_k - v_i) + 2\eta_h \varepsilon_k = 0,$$

$$\varepsilon_k = -\frac{\sum_{i=1}^c (u_{ki})^m (x_k - v_i)}{\eta_h + \sum_{i=1}^c (u_{ki})^m}.$$

From Karush-Kuhn-Tucker conditions,

$$\eta_h \frac{\partial L_{\text{sFCMMR}}}{\partial \eta_h} = \eta_h \left(\sum_{x_k \in M_h} \|\varepsilon_k\|^2 - (K_h)^2 \right) = 0.$$

Then, we have to consider two cases, $\eta_h = 0$ and $\sum_{x_k \in M_h} \|\varepsilon_k\|^2 - (K_h)^2 = 0$.

First, we consider $\eta_h = 0$, that means no constraints, i.e., $\sum_{x_k \in M_h} \|\varepsilon_k\|^2 < (K_h)^2$. In this case,

$$\varepsilon_k = -\frac{\sum_{i=1}^c (u_{ki})^m (x_k - v_i)}{\sum_{i=1}^c (u_{ki})^m}. \quad (7)$$

Second, we consider $\sum_{x_k \in M_h} \|\varepsilon_k\|^2 - (K_h)^2 = 0$.

$$\sum_{x_k \in M_h} \|\varepsilon_k\|^2 = \sum_{x_k \in M_h} \left\| -\frac{\sum_{i=1}^c (u_{ki})^m (x_k - v_i)}{\eta_h + \sum_{i=1}^c (u_{ki})^m} \right\|^2 = K_h^2,$$

$$\eta_h + \sum_{i=1}^c (u_{ki})^m = \pm \frac{\sqrt{\sum_{x_k \in M_h} \|\sum_{i=1}^c (u_{ki})^m (x_k - v_i)\|^2}}{K_h}. \quad (8)$$

The sign of the right side of (8) is plus from the constraint of η_h and hence,

$$\varepsilon_k = -\frac{K_h \sum_{i=1}^c (u_{ki})^m (x_k - v_i)}{\sqrt{\sum_{x_k \in M_h} \| \sum_{i=1}^c (u_{ki})^m (x_k - v_i) \|^2}}. \quad (9)$$

From (7) and (9),

$$\begin{aligned} \varepsilon_k &= -\alpha_k(m) \sum_{i=1}^c (u_{ki})^m (x_k - v_i), \\ \alpha_k(m) &= \min \left\{ \frac{1}{\sum_{i=1}^c (u_{ki})^m}, \frac{K_h}{\sqrt{\sum_{x_k \in M_h} \| \sum_{i=1}^c (u_{ki})^m (x_k - v_i) \|^2}} \right\}. \end{aligned}$$

Summarizing the above, we finally obtain the following optimal solutions:

$$v_i = \frac{\sum_{k=1}^n (u_{ki})^m (x_k + \varepsilon_k)}{\sum_{k=1}^n (u_{ki})^m}, \quad (10)$$

$$u_{ki} = \frac{(1/d_{ki})^{\frac{1}{m-1}}}{\sum_{i=1}^c (1/d_{ki})^{\frac{1}{m-1}}}, \quad (11)$$

$$(12)$$

$$\varepsilon_k = \begin{cases} -\alpha_k(m) \sum_{i=1}^c (u_{ki})^m (x_k - v_i), & (x_k \in M_h) \\ 0, & (x_k \notin M_h) \end{cases} \quad (13)$$

where

$$d_{ki} = \|x_k + \varepsilon_k - v_i\|^2, \quad (14)$$

$$\alpha_k(m) = \min \left\{ \frac{1}{\sum_{i=1}^c (u_{ki})^m}, \frac{K_h}{\sqrt{\sum_{x_k \in M_h} \| \sum_{i=1}^c (u_{ki})^m (x_k - v_i) \|^2}} \right\}. \quad (15)$$

we can construct the algorithm of sFCMMR using the optimal solutions obtained in the above.

Algorithm 1. Standard FCMMR

Step 1 Fix the values m , c and $\{M_h \mid 1 \leq h \leq r\}$. Set the initial values V and $\{\varepsilon_k \mid x_k \in M_h\}$. Set $\{\varepsilon_k \mid x_k \notin M_h\} = \{0\}$.

Step 2 Calculate U by the optimal solution (11) on fixing E and V .

Step 3 Calculate E by the optimal solution (13) on fixing V and U .

Step 4 Calculate V by the optimal solution (10) on fixing U and E .

Step 5 Finish the algorithm if the solutions are convergent, else go back to Step 2.

3.2 Entropy-Based FCMMR

Miyamoto et. al proposed entropy-based FCM (eFCM) in Ref. [8]. They constructed the algorithm of eFCM by introducing an entropy term $u_{ki} \log u_{ki}$ instead of the fuzzification parameter m of FCM into the objective function as follows:

$$J_{\text{eFCM}}(U, V) = \sum_{k=1}^n \sum_{i=1}^c (u_{ki} d_{ki} + \lambda^{-1} u_{ki} \log u_{ki}),$$

where λ is a constant and $d_{ki} = \|x_k - v_i\|^2$. The eFCM has an advantage that there is no singular point in comparison with sFCM.

In the same way as the above section, we can construct an algorithm of eFCM clustering with mutual relation constraints (eFCMMR). The objective function of eFCMMR is as follows:

$$J_{\text{eFCMMR}}(U, V, E) = \sum_{k=1}^n \sum_{i=1}^c (u_{ki} d_{ki} + \lambda^{-1} u_{ki} \log u_{ki}),$$

where $d_{ki} = \|x_k + \varepsilon_k - v_i\|^2$. The constraints are the same as sFCMMR. We obtain the following optimal solutions using the Lagrange multiplier.

$$v_i = \frac{\sum_{k=1}^n u_{ki} (x_k + \varepsilon_k)}{\sum_{k=1}^n u_{ki}}, \quad (16)$$

$$u_{ki} = \frac{e^{-\lambda d_{ki}}}{\sum_{i=1}^c e^{-\lambda d_{ki}}}, \quad (17)$$

$$\varepsilon_k = \begin{cases} -\alpha_k(1) (x_k - \sum_{i=1}^c u_{ki} v_i), & (x_k \in M_h) \\ 0, & (x_k \notin M_h) \end{cases} \quad (18)$$

where

$$d_{ki} = \|x_k + \varepsilon_k - v_i\|^2, \quad (19)$$

$$\alpha_k(1) = \min \left\{ 1, \frac{K_h}{\sqrt{\sum_{x_k \in M_h} \|x_k - \sum_{i=1}^c u_{ki} v_i\|^2}} \right\}. \quad (20)$$

We show the algorithm of eFCMMR in Algorithm 2.

Algorithm 2. Entropy-Based FCMMR

Step 1 Fix the values λ , c and $\{M_h \mid 1 \leq h \leq r\}$. Set the initial values V and $\{\varepsilon_k \mid x_k \in M_h\}$. Set $\{\varepsilon_k \mid x_k \notin M_h\} = \{0\}$.

Step 2 Calculate U by the optimal solution (17) on fixing E and V .

Step 3 Calculate E by the optimal solution (18) on fixing V and U .

Step 4 Calculate V by the optimal solution (16) on fixing U and E .

Step 5 Finish the algorithm if the solutions are convergent, else go back to Step 2.

4 Relation to Tolerance

The mutual relation and tolerance proposed in Refs. [10][11] are deeply connected, thus we describe the connection.

The concept of tolerance is a tool to handle uncertain data. This basic concept is simple. In general, a datum $x \in \Re^p$ with uncertainty is presented by some interval, i.e.,

$$[\underline{x}, \bar{x}] = [(\underline{x}_1, \dots, \underline{x}_p)^T, (\bar{x}_1, \dots, \bar{x}_p)^T] \subset \Re^p.$$

In our proposed tolerance, such a datum is represented by

$$\begin{aligned} x + \varepsilon &= (x_1, \dots, x_p)^T + (\varepsilon_1, \dots, \varepsilon_p)^T \in \Re^p \\ &= (x_1 + \varepsilon_1, \dots, x_p + \varepsilon_p)^T \end{aligned}$$

and a constraint for ε_j like that

$$|\varepsilon_j| \leq \kappa_j.$$

A vector $\varepsilon = (\varepsilon_1, \dots, \varepsilon_p)^T \in \Re^p$ is called tolerance vector. If we assume that

$$\begin{cases} x_j = \frac{\bar{x}_j + \underline{x}_j}{2}, \\ \kappa_j = \frac{|\bar{x}_j - \underline{x}_j|}{2}, \end{cases}$$

the formulation is equivalent to the above interval.

This concept of tolerance is very useful in the reason is that we can handle uncertain data in the framework of optimization to use the concept, without introducing some particular measure between intervals. For example, let's consider calculation of distance $d(X, Y)$ between $X = [\underline{x}, \bar{x}]$ and $Y = [\underline{y}, \bar{y}]$. We have to introduce some measure between intervals to calculate it, e.g.,

$$\begin{cases} d_{\min}(X, Y) = \min\{\|\underline{y} - \bar{x}\|, \|\bar{y} - \underline{x}\|\}, & \text{(minimum distance)} \\ d_{\max}(X, Y) = \max\{\|\underline{y} - \bar{x}\|, \|\bar{y} - \underline{x}\|\}, & \text{(maximum distance)} \\ d_{\text{Hausdorff}}(X, Y) = \max\{\|\bar{y} - \bar{x}\|, \|\underline{y} - \underline{x}\|\}. & \text{(Hausdorff distance)} \end{cases}$$

However, if we use tolerance, we don't need any particular distance, that is, a distance $d(X, Y)$ between $X = x + \varepsilon_x$ ($\|\varepsilon_x\| \leq \kappa_x$) and $Y = y + \varepsilon_y$ ($\|\varepsilon_y\| \leq \kappa_y$) can be calculated as $\|(x - y) + (\varepsilon_x - \varepsilon_y)\|$. From the above, we know that this tool is useful when we handle the data, especially data with missing values of their attributes, in the framework of optimization like as FCM [13]. From the above the tolerance was formulated as follows:

$$\|\varepsilon_k\|^2 \leq (\kappa_k)^2, \quad \forall k, \quad (1 \leq k \leq n) \quad (\kappa_k \geq 0) \quad (21)$$

where ε_k and κ_k are called a tolerance vector of x_k and a tolerance range of ε_k , respectively. The fuzzy c -means with the concept of tolerance is called fuzzy c -means for data with tolerance (FCMT) [10,11,12,13,14].

From the viewpoint of mutual relation, the tolerance is a class of mutual relation in which the element of M_h is just one, i.e., under the following condition:

$$M_k = \{x_k\}, \quad \forall k, \quad (22)$$

we can find the following relations:

$$\sum_{x_k \in M_h} \|\varepsilon_k\|^2 = \|\varepsilon_k\|^2 \leq (\kappa_k)^2 = (K_h)^2, \quad (K_h > 0) \quad (23)$$

$$\sum_{x_k \notin M_h} \|\varepsilon_k\|^2 = \|\varepsilon_k\|^2 \leq (\kappa_k)^2 = 0. \quad (24)$$

(5) and (23), and (6) and (24) are equivalent, respectively. Thus, we can replace (5) and (6) with (21). Consequently, we know that the FCMMR is equivalent to FCMT with the constraints (22).

5 Conclusion

In this paper, we proposed a new “semi-supervised” concept called mutual relation to formulate the trade-off relation and constructed two types of new clustering algorithms with the mutual relation constraints based on FCM (FCMMR). The discussions are in the framework of optimization. Moreover, we showed the connection between mutual relation and tolerance.

We had mainly theoretical discussions and we don't evaluate the abilities of sFCMMR and eFCMMR through numerical examples. In the forthcoming paper, we will evaluate the ability through some artificial and real numerical examples.

Acknowledgment. We would like to thank gratefully and sincerely Associate Professor KANZAWA Yuchi of Shibaura Institute of Technology and Processor MIYAMOTO Sadaaki of University of Tsukuba for their advice. This study is partly supported by the Grant-in-Aid for Scientific Research (C) (Project No.21500212) from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

1. Demiriz, A., Bennett, K., Embrechts, M.: Semi-Supervised Clustering using Genetic Algorithms. *Artificial Neural Networks in Engineering, ANNIE* (1999)
2. Wagstaff, K., Cardie, C., Rogers, S., Schroedl, S.: Constrained K-means Clustering with Background Knowledge. In: Proc. of the 8th International Conference on Machine Learning, pp. 577–584 (2001)
3. Basu, S., Banerjee, A., Mooney, R.J.: Semi-Supervised Clustering by Seeding. In: Proc. of 19th International Conference on Machine Learning (ICML 2002), pp.19–26 (2002)

4. Basu, S., Bilenko, M., Mooney, R.J.: Comparing and Unifying Search-Based and Similarity-Based Approaches to Semi-Supervised Clustering. In: Proc. of the ICML 2003 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining, pp. 42–49 (2003)
5. Chapelle, O., Schölkopf, B., Zien, A. (eds.): Semi-Supervised Learning. The MIT Press, Cambridge (2006)
6. Dunn, J.C.: A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics* 3, 32–57 (1973)
7. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York (1981)
8. Miyamoto, S., Mukaidono, M.: Fuzzy c -Means as a Regularization and Maximum Entropy Approach. In: Proc. of the 7th International Fuzzy Systems Association World Congress (IFSA 1997), vol. 2, pp. 86–92 (1997)
9. MacQueen, J.B.: Some Methods of Classification and Analysis of Multivariate Observations. In: Proc. of 5th Berkeley Symposium on Math. Stat. and Prob., pp. 281–297 (1967)
10. Endo, Y., Murata, R., Haruyama, H., Miyamoto, S.: Fuzzy c -Means for Data with Tolerance. In: Proc. 2005 International Symposium on Nonlinear Theory and Its Applications, pp. 345–348 (2005)
11. Murata, R., Endo, Y., Haruyama, H., Miyamoto, S.: On Fuzzy c -Means for Data with Tolerance. *Journal of Advance Computational Intelligence and Intelligent Informatics* 10(5), 673–681 (2006)
12. Kanzawa, Y., Endo, Y., Miyamoto, S.: Fuzzy c -Means Algorithms for Data with Tolerance based on Opposite Criterions. *IEICE Trans. Fundamentals* E90-A(10), 2194–2202 (2007)
13. Endo, Y., Hasegawa, Y., Hamasuna, Y., Miyamoto, S.: Fuzzy c -Means for Data with Rectangular Maximum Tolerance Range. *Journal of Advanced Computational Intelligence and Intelligent Informatics* 12(5), 461–466 (2008)
14. Endo, Y., Hasegawa, Y., Hamasuna, Y., Kanzawa, Y.: Fuzzy c -Means Clustering for uncertain Data using Quadratic Regularization of Penalty Vectors. *Journal of Advance Computational Intelligence and Intelligent Informatics* 15(1), 76–82 (2011)

Adaptive HTTP Request Distribution in Time-Varying Environment of Globally Distributed Cluster-Based Web System

Anna Zatwarnicka and Krzysztof Zatwarnicki

Department of Electrical, Control and Computer Engineering
Opole University of Technology, Opole, Poland

a.zatwarnicka@po.opole.pl, k.zatwarnicki@gmail.com

Abstract. A great development of Web technologies requires the application of complex systems and algorithms for maintaining high quality of Web services. The paper presents application of a broker-based architecture to the globally distributed Web system and the deployment of adaptive request distribution algorithm using neuro-fuzzy models in the decision-making process. The presented results of the simulation experiments show how the Web system operates under control of the proposed algorithm and the way of adaptation of the system to time-varying environment.

Keywords: cluster-based Web system, quality of Web service, request distribution, neuro-fuzzy models.

1 Introduction

Today's knowledge and content web-based technologies call for adaptive and intelligent algorithms for accessing the resources in globally and locally distributed Web systems composed by multiple Web servers. The most commonly used architecture of distributed Web systems is a cluster-based architecture where a set of Web and application servers operate collectively as a single Web resource in the network. The Web cluster includes a dispatcher that distributes user requests among Web servers located in the LAN. Further improvement of the Web performance needs application of distributed Web systems consisting of Web servers or clusters located at different geographical locations in the Internet.

This work presents the application of two machine learning techniques in an adaptive decision making framework, namely a fuzzy logic and neural networks, to deploy the adaptive and intelligent dispatching algorithm for resource requesting within a geographically distributed fully replicated Web sites. Fuzzy and neural systems have many applications in deployment of intelligent systems [2,7]. Fuzzy logic controllers can deal with the uncertainties that exist in physical systems [2]. By applying of neural networks into fuzzy systems we can achieve learning and adaptation characteristics.

The neuro-fuzzy models have already been applied to control operations of the locally distributed cluster-based Web systems. The most well-known applications are

the LFNRD [11] and the FARD [1] algorithms minimizing HTTP request service times and the AdaptLoad algorithm [10] distributing the requests on the base of the size of requested objects. There are not many applications of neuro-fuzzy models in control of globally distributed Web systems. One of them is GARDiB (Globally Aware Request Distribution with Broker) method present in this paper. We will describe the broker-based architecture, method and the distribution algorithm. Through the simulation experiments we will show how the algorithm distributes requests of different kinds according to the load and capacity of local clusters and the throughput of the network.

The rest of the paper is organized as follows. First, we introduce the idea of broker-based global request distribution. Then, we show the design of the Broker controlling operations of the Web system. After that, we discuss the simulation model and the results. Finally, we present the conclusion.

2 GARDiB Method and Algorithm

The GARDiB is an adaptive method enabling HTTP request distribution in the cluster-base globally distributed Web system. The GARDiB is not only the method describing way of operation of the system but also a distribution algorithm determining Web cluster to service the request. The GARDiB Web system operating in accordance to GARDiB method consist of following elements: Local Services (LS), Broker servers, DNS server and clients sending HTTP request. Fig. 1 presents the general schema of described system.

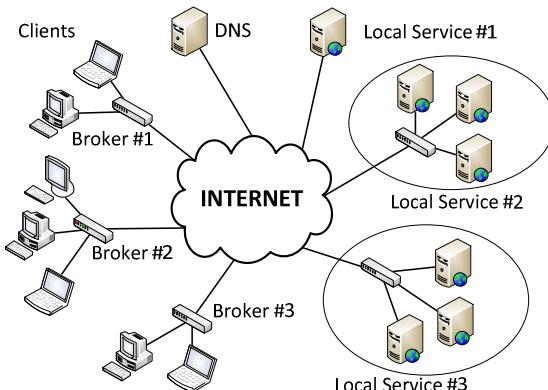


Fig. 1. Broker-based globally distributed Web system

The LS is a cluster of locally distributed Web servers or a single Web server. The LSs are distributed over the Internet and over the countries and he continents. We assume that the content offered in the service is fully replicated among the Local Services and the Web servers and each of the Web server can service each request belonging to the set of requests accepted in the service.

The Broker is placed between the clients and the LSs, at the edge of the Internet, nearby large concentration of clients. The Broker supervises the process of servicing HTTP requests. It receives requests send by the clients to the service, chooses the LSs to service requests, sends request to LSs, receives replays and transfers them to clients. The client can send requests to the LSs only through Brokers. There are many Brokers in the GARDiB Web system.

Clients in the system are applications (Web browsers, crawlers) generating HTTP requests. The DNS server is an integral part of the system. It redirects clients to the nearest Broker. Such solutions are known from literature [2] and will be not discussed further.

To complete servicing the HTTP request in the system following operations should be performed. At first the client gets through the DNS system the IP address of the nearest Broker, then the client sends the HTTP request to the Broker, which redirects it to the chosen LS. The LS service the request and sends the response to the Broker. The Broker transfers the response to the client.

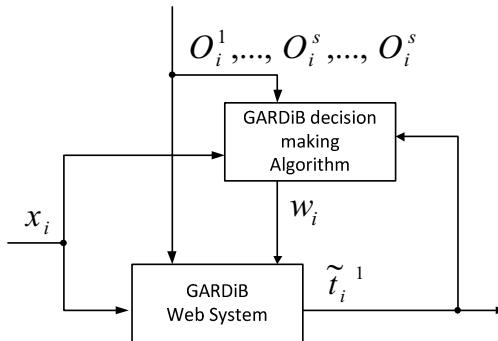


Fig. 2. Decision making process in the GARDiB system

The Broker is a key element of GARDiB system, controlling operations of the system in this way to minimize response times for each individual HTTP request. The construction of the device and the request distribution algorithm can have significant impact on the response times. The request response time is measured from the moment the Broker takes the decision concerning allocation of the request, to the moment the response prepared by the LS is be transferred completely to the Broker. The request response time consists in the transition time of the requests and the response, and the time of servicing the request in the LS.

Let us introduce following denotations necessary to formulate the problem: x_i – the HTTP request, $x_i \in X$, where X is the set of request serviced in the Web system; i – the index of the request $i = 1, 2, 3, \dots$; \tilde{t}_i – the i th request response time; \hat{t}_i^s – the estimated response time for the i th request and the s th LS, $s = 1, \dots, S$; S – number of LSs; w_i – the decision, the LS chosen to service the i th request, $w_i \in \{1, \dots, S\}$; O_i^s – the load of the s th LS at the i th moment.

According to our conception the Broker chooses for each incoming request x_i the local service w_i in this way that $w_i = \min_s \{\hat{t}_i^s : s = 1, \dots, S\}$, taking into account response times, estimated on the base of knowledge of the load O_i^s , $s = 1, \dots, S$, of LSs and the knowledge of past measured request response times $\tilde{t}_1, \dots, \tilde{t}_{i-1}$. The schema of the decision process in the GARDiB system is presented in the Fig. 2.

2.1 Design of the GARDiB Broker

The GARDiB Broker consists in three main modules (see Fig. 3): the decision module choosing the LS to service the request, the execution module responsible for execution of the decision and the measurement module collecting information necessary in the decision process.

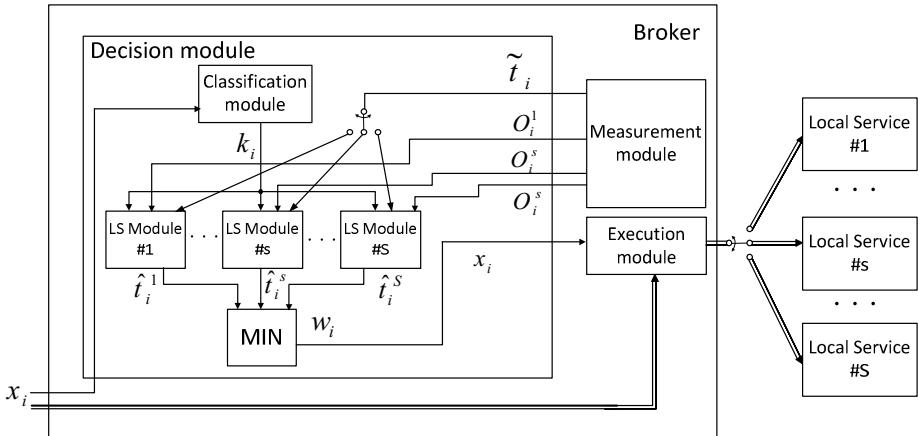


Fig. 3. GARDiB Broker

The decision module implements the GARDiB distribution algorithm according to which the classification module, the LS models and the MIN block can be distinguished.

The classification module classifies incoming requests. The class k_i of the request is determined on the base of the requested object's size, in the case of static objects where every dynamic object has its own individual class, $k_i \in \{1, \dots, K\}$ and K is the number of classes. Objects belonging to the same class have similar service times.

Each model of LS in the decision module is assigned to one LS in the GARDiB system. After the HTTP request arrival the model estimates the request response time \hat{t}_i^s taking into account the load O_i^s and the class k_i of the request. The load is measured by the measurement module and $O_i^s = [a_i^s, b_i^s]$, where a_i^s is a number of requests serviced simultaneously by s th LS and b_i^s is the transmission time of the

probe object downloaded periodically from LS by the measurement module. The probe objects are prepared by the LSs and contains information concerning the load of the LS.

Estimating the request service times the model of the LS takes into account not only the load of s th LS but also the situation in the WAN on the way between the LS and the Broker. The construction of the LS model will be described in the farther part of the paper.

The MIN block takes decision w_i choosing LS offering the shortest response time $w_i = \min_s \{\hat{t}_i^s : s = 1, \dots, S\}$, $w_i \in \{1, \dots, S\}$.

The execution module supervises the process of sending the request to the chosen LS, receives the response and transfers it the client.

The measurement module measures the request service time \tilde{t}_i and transfers it to the LS model which correspond to the LS servicing the i th request. The module also measures periodically the load O_i^s , $s = 1, \dots, S$.

2.2 Local Service Model

The LS model corresponds to the real LS operating in the Web service. Each model includes: a neuro-fuzzy model of the LS and a Database Unit U_i^s storing parameters of the neuro-fuzzy model. The LS model estimates the response time \hat{t}_i^s on the base of the class k_i and the load a_i and b_i . After finishing servicing of the request by the s th LS the parameters U_i^s are tuned taking into account the measured request response time \tilde{t}_i . The neuro-fuzzy model of the LS is presented in the Fig. 4.a. For clarity of denotations the index s of the server will be dropped in the denotations in the description of neuro-fuzzy model.

In the first input layer of neurons in the neuro-fuzzy model the values of the membership to individual input fuzzy sets are calculated. The fuzzy sets for the input a_i are denoted as $Z_{a1}, \dots, Z_{al}, \dots, Z_{aL}$, similarly fuzzy sets for the input b_i are $Z_{b1}, \dots, Z_{bm}, \dots, Z_{bM}$. Membership functions for inputs are denoted as follow $\mu_{Z_{al}}(a_i)$, $\mu_{Z_{bm}}(b_i)$, $l = 1, \dots, L$, $m = 1, \dots, M$. The membership functions are triangular (see Fig. 4.b) and the parameters $\alpha_{1ki}, \dots, \alpha_{lki}, \dots, \alpha_{(L-1)ki}$ (for input a_i) and $\beta_{1ki}, \dots, \beta_{mki}, \dots, \beta_{(M-1)ki}$ (for input b_i) specify the shape of functions. It has been assumed that membership functions for all input fuzzy sets are triangular. The functions are piece-wise linear and have a limited support and therefore the process of calculating the degrees of membership is not time-consuming. Also the shape of the function is described with a low number of parameters, which may be tuned in the adaptation process. In addition, the author's experience strength strongly the understanding of the linguistic space in terms of the triangular membership functions.

The fuzzy sets for the output are denoted as $T_1, \dots, T_j, \dots, T_J$ and the membership functions for output are singletons indicating values $t_{1ki}, \dots, t_{jki}, \dots, t_{Jki}$ (Fig. 4.c).

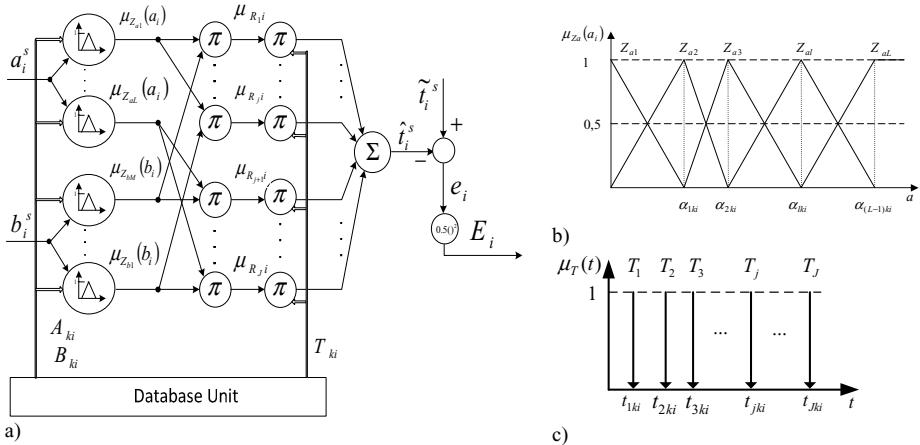


Fig. 4. a) Neuro-fuzzy model of LS, b) Membership functions for input, c) Membership functions for output

The parameters of membership functions are stored in Database Unit for each class individually $U_i = [U_{1i}, \dots, U_{ki}, \dots, U_{Ki}]^T$, where $U_{ki} = [A_{ki}, B_{ki}, Y_{ki}]$ and $A_{ki} = [\alpha_{1ki}, \dots, \alpha_{lki}, \dots, \alpha_{(L-1)ki}]$, $B_{ki} = [\beta_{1ki}, \dots, \beta_{mki}, \dots, \beta_{(M-1)ki}]$, $T_{ki} = [t_{1ki}, \dots, t_{jki}, \dots, t_{Jki}]$.

In the second layer of neurons the products of inference are obtained for each fuzzy set of the inputs a_i and b_i combined together $\mu_{R_{ji}} = \mu_{Z_{al}}(a_i) \cdot \mu_{Z_{bm}}(b_i)$, $j = 1, \dots, J$, $J = M \cdot L$.

In the third layer the estimated service time value is calculated with use of the Height Method $\hat{t}_i = \sum_{j=1}^J t_{jki} \mu_{R_{ji}}$.

In the process of adaptation the parameters of membership functions for inputs and output are tuned. The back propagation algorithm and gradient descent rule are used to tune the parameters according to $\delta_{(i+1)} = \delta_i - \eta \partial E_i / \partial \delta_i$, where $\delta_{(i+1)}$ is a new value of δ_i , η is learning rate, $\partial E_i / \partial \delta_i$ is partial error and $E_i = (e_i)^2 / 2$ and $e_i = \hat{t}_i - \tilde{t}_i$.

New values of parameters are calculated as follow $t_{j,k(i+1)} = t_{jki} + \eta_t \mu_{R_{ji}} (\tilde{t}_i - \hat{t}_i)$, $\alpha_{l,k(i+1)} = \alpha_{lki} + \eta_a (\tilde{t}_i - \hat{t}_i) \sum_{\gamma=1}^M \left(\mu_{Z_{b\gamma}}(b_i) \sum_{\varphi=1}^L \left(t_{((m-1)L+\varphi)ki} \partial \mu_{Z_{a\varphi}}(a_i) / \partial \alpha_{lki} \right) \right)$,

$$\beta_{m k(i+1)} = \beta_{m k i} + \eta_b (\tilde{t}_i - \hat{t}_i) \sum_{\varphi=1}^L \left(\mu_{Z_{a\varphi}}(a_i) \sum_{\gamma=1}^M \left(t_{((l-1)M+\gamma)ki} \frac{\partial \mu_{Z_{b\gamma}}(b_i)}{\partial \beta_{m k i}} \right) \right),$$

where η_y , η_a and η_b are adaptation ratios which values determined in preliminary experiments are $\eta_y = 0.5$, $\eta_a = \eta_b = 0.01$.

3 Simulation Model and Experiment Results

The Web system working under control of the GARDiB algorithm was evaluated through simulations experiments. The simulation program was built with the use of the CSIM19 package [5]. The simulator included the following modules: the request generator, the Internet, the Broker and the Local Service modules (see Fig. 5.a).

The request generator module was working in the way that the generated request traffic complied with the traffic observed on the Internet, which is characterized by bursts and self-similarity [6] (Fig. 5.b). The 80% of requests were static requests serviced by Web servers and 20% of requests were dynamic requests serviced by Web and database servers.

The Internet module modeled the latency concerned with transition of the request and the response in the Internet. The transmission time was calculated as follow

$$transmission_time = RTT + \frac{object_size + HTTP_header_size}{throughput}$$

[9]. RTT is the Round

Trip Time measured between the source and the sink. The *HTTP_header_size* is the size of HTTP response header (the average size is 290B). Effective throughput is the number of bytes transmitted successfully in the time unit. The RTTs and the effective throughput values were obtained during special experiments provided with use the wget software and a dedicated software collecting values of RTT and effective throughput in the time intervals.

The document RFC 1945 (47 kB) was downloaded from academic centers characterized by a broadband Internet connection and low load of Web servers. Following centers were chosen: the SURFnet in the Netherlands, the Australian National University in Australia and Massachusetts Institute of Technology in USA.

Requests created by the generator were directed through the Internet module to the observed Broker module and directly to LS modules simulating in this way operations of other Brokers.

The GARDiB algorithm as well as other popular algorithms used in industrial solutions were implemented in the Broker: Round Robin algorithm (RR) and two versions of the Weighted Round Robin algorithm, namely Weighted Round-Robin Load (WRR_L) (the weights are determined on the base of the LS workload rate, measured as the number of requests processed concurrently) [8] and the Weighted Round-Robin Round Trip Time (WRR_T) (where the weights are determined on the basis of the RTT time) [4].

Each LS model consisted of the Web switch and the Web and the database servers. The Web servers were modeled as queuing networks composed of the processor and hard drive resources. The service times for the processor were as follow [2]: the TCP connection establishment and the teardown operation cost 14.5 μ s, the transmit processing 4.0 μ s per 512 bytes. Adopted service times for hard drive: the disk

transfer time 41.0 μ s per 4 kB, the seek time 2.8 ms, the seek time in case of big files additional 1.4 ms for every 44 kB. The cache memory was operating according to the Least Recently Used replacement policy.

The database servers were composed of the single queues [3] with service times modeled according to hyperexponential distribution (see Fig. 5.c). The Web switch controlling operations of LS was distributing HTTP requests according to the LFNRD policy minimizing request response times.

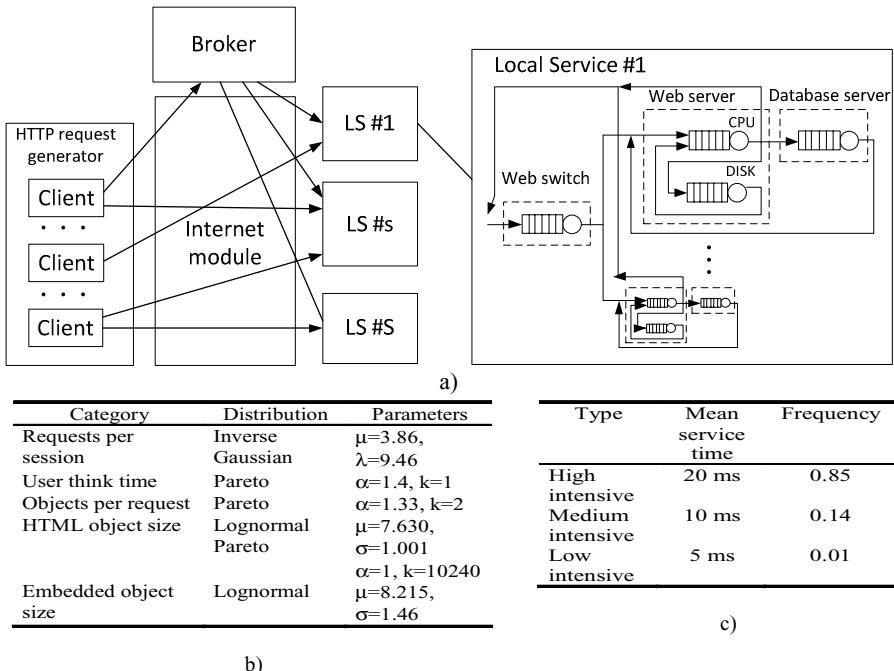


Fig. 5. a) A simulation model; b) Workload model parameters; c) Workload model parameters of dynamic objects

Experiments have been conducted for two different configurations of LSs and the load. In both experiments the LSs were placed in the Netherlands, USA and Australia. The Broker was placed in Poland. The best transfer rates were between the Broker and the LS in the Netherlands while the worst with the LS in Australia. In the first configuration each LS consisted of three Web and backend servers and 25% of the load was redirected to each LS and to the Broker (configuration denoted as 3NL25/3USA25/3AU25/Broker25).

In the second configuration the LS in the Netherlands consisted of 2 Web and backend servers, the LS in USA had 3 sets of servers while the LS in Australia had 1 set. 20% of the load was directed to LS in the Netherlands, 30% to USA, 10% to Australia and 40% to the Broker (configuration denoted as 2NL20/3USA30/1AU10/Broker40).

In the Fig 6.a and 6.b the request response time in the load (number of new clients per second) function for different request distribution algorithms is presented. The Fig. 6.c and 6.d presents the diagrams of percentage of dynamic requests serviced by individual LSs.

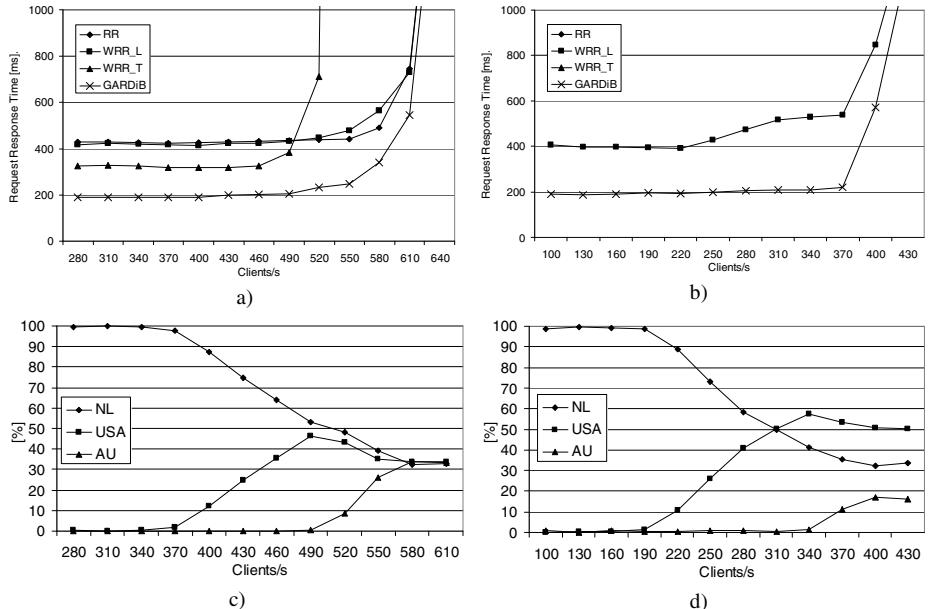


Fig. 6. Request response time vs. load in a) 3NL25/3USA25/3AU25/Broker25, b) 2NL20/3USA30/1AU10/Broker40; Distribution of dynamic requests among LSs in c) 3NL25/3USA25/3AU25/Broker25, d) 2NL20/3USA30/1AU10/Broker40

As one can see the shortest request response times were obtained for the GARDiB algorithm. In the experiment where the number of Web servers for each LS were different only the GARDiB and the WRR_L algorithms were operating well obtaining acceptable response times. In each of the experiments 100% of static requests were serviced by the LS in the Netherlands characterized by the best transfer rate. Analyzing the Fig. 6.c and 6.d it is noticeable that in case of low load 100% of dynamic requests were also serviced by the LS in the Netherlands while when the load was very high the dynamic requests were redirected even to LSs with poor transfer rates.

The number of requests serviced by individual LSs in case of high load is proportional to number of Web servers possessed by the LSs. Results of the experiments show that the Web system operating under control of the GARDiB algorithm can adapt to the time-varying environment changing the strategy of operations.

4 Summary

In this paper the problem of request distribution in the globally distributed Web system was discussed. We presented the broker-based architecture of the Web system used in our GARDiB method and the adaptive request distribution algorithm. Presented results of simulation experiments show that the Web system operating under control of the GARDiB algorithm can achieve short request response times and that the system adopts to the time-varying environment. In the future we are going to evaluate the proposed method in empirical experiments.

References

1. Borzemski, L., Zatwarnicki, K.: A fuzzy adaptive request distribution algorithm for cluster-based Web Systems. In: Proc. of the Eleventh EUROMICRO Conference on Parallel, Distributed and Network-Based Processing, pp. 119–126. IEEE Computer Society Press, Los Alamitos (2003)
2. Borzemski, L., Zatwarnicki, K., Zatwarnicka, A.: Adaptive and Intelligent Request Distribution for Content Delivery Networks. *Cybernetics and Systems* 38(8), 837–857 (2007)
3. Casalicchio, E., Tucci, S.: Static and dynamic scheduling algorithm for scalable Web server farm. In: Proc. of 9th Euromicro Workshop on Parallel and Distributed Processing (PDP 2001), Mantova, Italy, pp. 369–376 (2001)
4. Cisco Boomerang, http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/12_4/nm_12_4_book.html
5. CSIM19, http://www.mesquite.com/products/CSIM19_datasheet.pdf
6. Crovella, M.E., Bestavros, A.: SelfSimilarity in World Wide Web traffic evidence and possible causes. In: SIGMETRICS 1996, Philadelphia, USA, pp. 835–846 (1996); IEEE/ACM Transactions on Networking 5(6) (1997)
7. Gilly, K., Juiz, C., Puigjaner, R.: An up-to-date survey in web load balancing. Springer, Heidelberg (2010), *World Wide Web*, 10.1007/s11280-010-0101-5
8. IBM RedBooks, <http://www.redbooks.ibm.com/abstracts/TIPS0235.html>
9. Menasse, D.A., Almeida, V.A.F.: Capacity planning for Web Performace. Metrics, Models, and Methods. Prentice Hall, New York (2002)
10. Riska, A., Wei, S., Smirni, E., Ciardo, G.: ADAPTLOAD: effective balancing in clustered w1. b servers under transient load conditions. In: Proceedings of 22nd International Conference on Distributed Computing Systems, Vienna, Austria, pp. 104–111. IEEE Computer Society, Los Alamitos (2002)
11. Zatwarnicki, K.: Application of fuzzy-neural models in the distribution of HTTP requests in the local Web server cluster. *Information Systems Architecture and Technology*. In: Advances In Web-Age Information Systems, pp. 47–57. Wroclaw University of Technology Press, Wroclaw (2009)

Distributed BitTable Multi-Agent Association Rules Mining Algorithm

Walid Adly Atteya¹, Keshav Dahal¹, and M. Alamgir Hossain²

¹ School of Computing, Informatics and Media, Bradford University, United Kingdom
`{waaabdo, k.p.dahal}@brad.ac.uk`

² School of Computing, Engineering and Information Sciences, Northumbria University,
United Kingdom
`alamgir.hossain@northumbria.ac.uk`

Abstract. Many algorithms have been proposed for the discovery of association rules. The efficiency of these algorithms needs to be improved to handle real-world large datasets. This efficiency can be determined mainly by three factors. The way candidates are generated, the way their supports are counted and the data structure used. Most papers focus on the first and the second factors while few focus on the underlying data structures. In this paper, we present a distributed Multi-Agent based algorithm for mining association rules in distributed environments. The distributed MAS algorithm uses Bit vector data structure that was proved to have better performance in centralized environments. The algorithm is implemented in the context of Multi-Agent systems and complies with global communication standard Foundation for Intelligent Physical Agents (FIPA). The distributed Multi-Agent based algorithm with its new data structure improves implementations reported in the literature that were based on Apriori. The algorithm has better performance over Apriori-like algorithms.

Keywords: Multi-Agent Systems, Distributed Data Mining, Association Rules.

1 Introduction

Finding frequent itemsets is one of the most important data mining research fields. The problem was first presented in [1] with another extension in [2]. Its main algorithm, Apriori, had an impact on other data mining techniques as well. Association rules and frequent itemsets mining became a widely research area, and hence, most researchers have tried to present faster algorithms. Many of these algorithms were Apriori-based or Apriori extensions. Most association rule algorithms use hash-trees extensively to speed up the search for itemsets. Those who adopted Apriori strategy tended to adopt the whole set of procedures and data structures as well.

Recently, algorithms have been proposed to increase the efficiency of these algorithms to improve real-world large datasets. Some algorithms focused on the way candidates are generated. Others focused on the way their supports are counted. Few researchers have focused on the underlying data structure used which was a hash-tree in case of Apriori-based algorithms.

Park et al. [9] has invented a well known technique called DHP (Direct Hashing and Pruning) and was enhanced in [10]. DHP uses a hash technique that makes it very

efficient for the generation of candidate itemsets, in particular for the large two-itemsets and employs effective pruning techniques. The reduction of the number of generated candidates greatly improves the performance of the whole process. However, Park used this hashing technique to mine association rules in centralized database. Bodon [6] has demonstrated that a Trie data structure outperforms hash-trees. Tries appeared to offer simpler and scalable algorithms which turned out to be faster. Bodon has implemented Apriori association rule mining algorithm using Trie data structure rather than Hash Tree. Further publication [3] proved that the data structure Trie appeared to be faster than the original algorithm. Bodon has extended his implementation for mining itemset sequences in [5]. Other researchers have adopted the Trie structure to mine association rules on centralized databases [4].

Recently, a novel approach by Dong has presented a very effective algorithm named as BitTableFI [7]. The algorithm uses a special data structure BitTable horizontally and vertically to compress database for quick candidate itemsets generation and support count, respectively. Dong has proven that this data structure is faster than the hash tree used by Apriori. Results were obtained by applying the BitTable data structure on two synthetic centralized datasets. Song et al. [11] is one of the extensions that is based on this technique.

In this paper, we present an efficient distributed MAS algorithm. The efficiency of the algorithm is obtained by modifying the data structure used and the way candidates are generated and counted. The rest of the paper is organized as follows. The next section describes the proposed distributed BitTable Multi-Agent based algorithm. Section 3 describes the model experiments and evaluation. The last section presents the conclusion and the future work.

2 Distributed BitTable Multi-Agent Association Rules Algorithm

In earlier work, we have presented an enhancement for Apriori algorithm using a simpler data structure [8]. The algorithm was implemented on centralized database. Previous work has extended the basic concepts of Apriori like algorithms to work in distributed environments using cooperative Multi-Agents [12]. The parallelism of the candidate generation and the support count processes among these distributed agents helped in decreasing the time needed for the whole mining process. The previously proposed algorithm was implemented on distributed medical databases [13] for patient diagnostic system regarding Inflammation of urinary bladder and Nephritis of renal pelvis origin diseases. The proposed model improved the diagnostic knowledge and discovered the diseases based on the minimum number of effective tests, thus, provided accurate medical decisions based on cost effective treatments. The constructed Knowledge base could predict the existence or the absence of the diseases, thus improving the medical service for the patients.

In this section, we present the distributed BMAS algorithm which combines the best of different association rules algorithms and techniques in order to achieve the best performance and execution time. The proposed algorithm combines the association rules as a data mining technique, the BitTable data structure that was proved to be a very efficient data structure for mining frequent itemsets [11] [14] and the Multi-Agents technique to decrease the time needed for the candidate generation and the support count

processes [12]. Databases with this structure are very compressed and can easily fit in memory. Moreover, it has a great effect on the performance in the candidate generation phase and the support counting which are the most lengthy processes in the frequent itemsets generation algorithms. Dong has implemented his BitTableFI algorithm on centralized database only. For some reasons which were not clearly stated in the paper, the BitTableFI algorithm constructs the frequent itemsets BitTable data structure after the frequent 2-itemsets are generated. We believe that constructing the frequent itemsets BitTableFI from the first iteration will increase the performance of the candidate generation and counting for the first and the second itemsets. Consequently, this will have a good impact on the algorithm performance. The proposed algorithm is described as follows:

1. The proposed algorithm is based on the distributed Multi-Agent based algorithm described in [12] which was based on Apriori like algorithms.
2. The proposed algorithm uses the BitTable data structure in [7] instead of the previously implemented data structure in [12].
3. Unlike the BitTableFI algorithm, the proposed algorithm constructs the BitTable data structure before the first iteration.
4. Unlike the BitTableFI algorithm which was implemented on two synthetic centralized datasets, the proposed algorithm together with the BitTableFI algorithm are to be implemented and tested on five distributed real world benchmark datasets.
5. The distributed Multi-Agent based algorithm complies with the global standards in communication between agents, namely the FIPA, thus enabling the ability for cooperating with other standard agents also the future extension for the proposed model.

2.1 The Database Conversion Algorithm

This section describes how local databases are converted into the BitTables format instead of the Apriori format. Every item is checked for existence in the transaction. If the item exists, the item is replaced with 1 otherwise it is replaced with 0. For instance if we have the items ABCDE and the transaction ACD, the bitvector representation for the transaction is 10110. The conversion of the database into the BitTables format is described in details in Algorithm 1

Let $I = \{i_1, i_2, \dots, i_n\}$ be a finite set of items and D be a dataset containing N transactions, where each transaction $t \in D$ is a list of distinct items $T = \{i_1, i_2, \dots, i_j\}$ where $i_j \in I (1 \leq j \leq |t|)$, and each transaction can be identified by a distinct identifier tid.

2.2 The Distributed BitTable MAS Algorithm

This section presents the proposed distributed BitTable Multi-Agent association rules algorithm. The algorithm consists of three types of cooperative agents that work together to achieve the required goals. The first kind of agents is the Interface Agent. This agent can cooperate with the human and accepts the user required support. The second kind of agents is the local agents which are distributed at local sites for generating the candidate itemset. The third kind of agents is the main agent which has a global view of

Algorithm 1. Transaction conversion into bit vectors (local agents)

```

begin
    Let  $\{I\}$  = the finite set of all items  $\{i_1, i_2, \dots, i_n\}$ ;
    foreach item  $i_j$  in the set of items  $\{I\}$  where ( $1 \leq j \leq n$ ) do
        if  $i_j = i_k$  where  $i_k \in t = \{i_1, i_2, \dots, i_{|t|}\}, i_k \in I, (1 \leq k \leq |t|)$  then
            |
            |    $t_{bitvector} += 1;$ 
            |
            |   else
            |       |
            |       |    $t_{bitvector} += 0;$ 
            |
            |   end
        |
        end
    |
    Output the database transactional bit vector  $t_{bitvector};$ 
end

```

all local agents and is responsible for the candidate generation process. All agents work together to achieve the required association rules mining goal. The proposed algorithm is compliant to the Foundation for Intelligent Physical Agents standard.

The proposed algorithm is described as follows:

1. The Interface Agent accepts the support from the user.
2. The Interface Agent sends the required support to the main agent.
3. Main Agent sends a "propose performative" FIPA message to Local Agents:

(Propose

:sender (agent-identifier :name main_agent)
 :receiver (set (agent-identifier :name local_agent))
 :content "Start mining with support = minsupp"
 :reply-with start_mining_proposal)

4. Local Agents reply with an "agree performative" to Main Agent as follows:

(Agree

:sender (agent-identifier :name local_agent)
 :receiver (set (agent-identifier :name main_agent))
 :content "proposal approved and mining started at k=1"
 :in-reply-to start_mining_proposal)

5. Each Local Agent starts counting the local supports for all 1- candidate itemsets in its local database according to its local number of records. **Algorithm 2 explains the counting process in details.**

6. Local Agent replies with "inform performative" to Main Agent as follows:

(Inform

:sender (agent-identifier :name local_agent)
 :receiver (set (agent-identifier :name main_agent))
 :content "finished counting candidate 1-itemsets")

7. Main Agent compares the summation of the local supports sent from all agents for 1-candidate itemsets with the min support supplied by the user.

8. Main Agent finds the 1-large itemsets and save it in the database in the list of frequent itemsets.

9. Main Agent sends an "Inform performative" FIPA message to Local Agents:

(Inform

:sender (agent-identifier :name main_agent)
 :receiver (set (agent-identifier :name local_agent))
 :content "frequent itemsets at k=1 are successfully generated")

10. Main Agent generates the k-candidate itemsets. **Candidate generation process is explained in details in Algorithm 3**

11. Main Agent sends the generated k-candidate itemsets to all local agents.

12. Main Agent sends a "Request performative" FIPA message to all Local Agents:

(Request

:sender (agent-identifier :name main_agent)
 :receiver (set (agent-identifier :name local_agent))
 :content "candidates are generated at iteration K, please count the support"
 :reply-with iteration_k)

13. Each Local Agent calculates the k-candidate itemsets in its local databases. **This can be explained in details in Algorithm 4**

14. Local Agents send an "Inform performative" FIPA message to Main Agent:

(Inform

:sender (agent-identifier :name local_agent)
 :receiver (set (agent-identifier :name main_agent))
 :content "finished counting candidate itemsets for the current iteration K"
 :in-reply-to iteration_k)

15. The Main Agent considers any k-candidate itemset as frequent if the summation of all local supports for this itemset from all local agents is greater than the min global support

16. Frequent itemsets are saved in the central database in the list of k-frequent itemsets while small itemsets are not considered in the next iteration.

17. Steps (10) to (16) are iterative and finish when there are no more k+1 candidate itemsets.

18. Main Agent sends an "Inform performative" message to all Local Agent :

(Inform

:sender (agent-identifier :name main_agent)
 :receiver (set (agent-identifier :name local_agent))
 :content "Finished mining of frequent itemsets")

19. Main Agent sends all frequent itemsets to Interface Agent for representation.

2.3 The Proposed Algorithm at the First Iteration

The generation of candidate itemsets and the large itemsets counting in the early iterations (k=1 and 2) are considered as the most time consuming processes for the overall association rules mining process. Unlike the BitFI algorithm, we apply the BitTableFI data structure starting from the first phase. This has significantly resolved the performance bottleneck especially for the first two iterations. Algorithm 2 presents how the 1-frequent itemsets are counted at distributed sites by local agents.

Algorithm 2. Counting the 1-frequent itemsets (local agents)

```

begin
  foreach item bit vector  $Ib_i$  in the items bit table do
    foreach transaction in the database do
      | Perform BitWise AND operation with  $t_{bitvector}$ ;
      | if  $Ib_i AND t_{bitvector} = Ib_i$  then
      |   | Increment the support( $Ib_i$ );
      | end
    end
    if total support ( $Ib_i$ )  $\geq minsupp$  then
      | Add  $Ib_i$  to set of large itemsets  $Lb_1$ ;
    end
  end
  Output set of 1-Frequent itemsets;
end

```

2.4 The Proposed Algorithm at the K-Iteration

Itemsets counted by local agents are sent to the main agent which generates the k-frequent itemset and the (k+1) candidate itemsets for the next iteration. The generation of the (k+1) candidate itemsets is described in Algorithm 3.

Algorithm 3. (k+1) candidate itemsets generation (main agent)

```

begin
  foreach frequent k-itemset bit vector  $F_{bitvector}^i$  in the set of frequent itemsets bit
  table( $F_{bk}$ ) do
    Get the Mid of  $F_{bitvector}^i$  = (set of items with the last bit = 1 changed to 0);
    foreach Frequent k-itemsets  $F_{bitvector}^j$  where
    ( $i + 1 \leq j \leq number\ of\ frequent\ k - itemsets$ ) do
      | Perform BitWise AND operation with  $F_{bitvector}^j$ ;
      | if Mid of  $F_{bitvector}^i$ . AND  $F_{bitvector}^j$  = Mid of  $F_{bitvector}^i$  then
      |   | Generate Candidate k-itemset bit vector  $C_{bitvector}^{k+1} = F_{bitvector}^i$  OR
      |   |  $F_{bitvector}^j$ ;
      |   | Add  $C_{bitvector}^{k+1}$  to the set of candidate k+1 itemsets bit table ( $Cb_{k+1}$ );
      | end
    end
  end
  Output set of Candidate  $k + 1$  itemsets bit table ( $Cb_{k+1}$ );
end

```

The generated candidate itemsets are sent to local agents which count their supports and send them back to the main agent. Algorithm 4 explains the counting process for the k-candidate itemsets.

Algorithm 4. Counting the k-frequent itemsets (local agents)

```

begin
  foreach candidate k-itemset bit vector  $C_{bitvector}^i$  in the candidates bit table ( $C_k$ ) do
    foreach transaction in the database do
      | Perform BitWise AND operation with  $t_{bitvector}$ ;
      | if  $C_{bitvector}^i$  AND  $t_{bitvector} = C_{bitvector}^i$  then
      |   | Increment the support( $C_{bitvector}^i$ );
      | end
    end
    if total support( $C_{bitvector}^i$ )  $\geq$  minsupp then
      | Add  $C_{bitvector}^i$  to set of large itemsets  $Lb_k$ ;
    end
  end
  Output set of k-Frequent itemsets;
end

```

3 Model Experiments and Evaluation

The experiments included the implementation and testing of four algorithms against five different real world datasets at five different supports with total of 100 readings. Two of the implemented algorithms were centralized (PEA [8] and BT [7]). The other two algorithms were distributed (MAS [12] and our proposed distributed BitTable Multi-Agent based algorithm BMAS). The five benchmark datasets from UCI machine learning repository are related to different application domains. Datasets are described in details in Table 1. For distributed algorithms, datasets were distributed almost equally in two different sites. The test bed used was windows XP, Intel Pentium IV processor, 2 gig ram. The results obtained are as follows.

Table 1. UCI Benchmark Datasets

Dataset	Number of instances	Number of attributes	Year
Abalone	4177	8	1995
Car Evaluation	1728	6	1997
Mammographic Mass	961	6	2007
Blood Transfusion Service Center	748	5	2008
Iris	150	4	1988

From Figures 1 to 5 we can observe the following:

1. In case of the centralized algorithms, BT algorithm with the BitTableFI data structure outperforms PEA algorithm.
2. In case of the distributed algorithms, the proposed BMAS with the BitTableFI data structure outperforms the previously implemented algorithm MAS.
3. The distributed algorithms BMAS and MAS outperform the centralized algorithms BT and PEA.

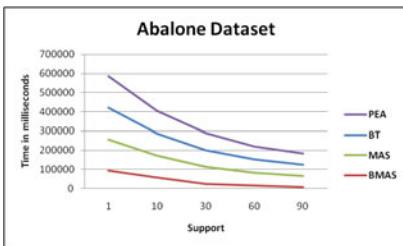


Fig. 1.

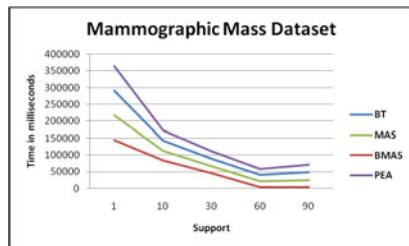


Fig. 2.

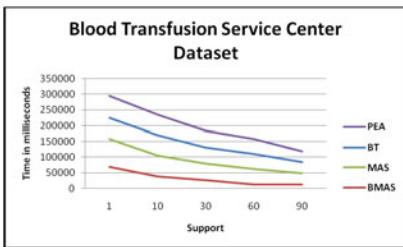


Fig. 3.

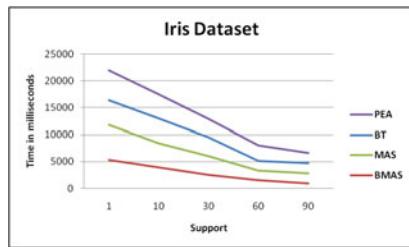


Fig. 4.

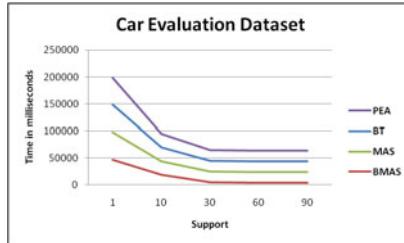


Fig. 5.

4. The proposed algorithm BMAS outperforms BT, PEA and MAS algorithms.
5. Although low support values are considered as one of the limitations for most association rules algorithms including BitTableFI and our proposed algorithm. However, Figures 1 to 5 show that the difference in execution time increases when the support value decreases. This shows that the proposed distributed BMAS algorithm outperforms the BitTableFI algorithm at low supports.

The performance of BMAS has been achieved due to the following reasons:

1. The use of the Bitwise And/Or operation to generate candidate itemsets based on BitTables data structure which was proved to be greatly faster than the traditional item comparing method used in many Apriori-like algorithms [6].
2. The highly compressed BitTable database constructed which helps in quick counting for the support of the candidate itemsets using the Bitwise And operation.
3. The construction of the BitTable data structure before the first iteration.

4. The use of the distributed Multi-Agents which decrease the time needed for candidate generation and support counting.

4 Conclusion

The efficiency of association rules algorithms needs to be improved to handle real-world large datasets. To increase the overall efficiency of the mining process, we presented a distributed Multi-Agent based algorithm to enhance the three main factors affecting the overall efficiency. First, to improve the way candidates are generated, our distributed algorithm is based on BitTable data structure which has a better performance in the candidate generation phase. Second, to improve the way by which candidate supports are counted, we have implemented distributed agents in local sites that help in the counting process. Third, we have implemented the BitTable data structure which helps in compressing the database thus can easily fit in memory at local sites. The BitTable data structure was implemented before the first iteration and not like the BitTableFI algorithm after the second iteration. This had a great impact on the algorithm performance. The distributed BitTable Multi-Agent based algorithm complies with the global standards in communication between agents, namely the FIPA, thus enabling the ability for cooperating with other standard agents also the future extension for the proposed model. Unlike the BitTableFI algorithm which was tested on two synthetic centralized datasets, the performance of algorithms was tested against five different real world datasets from UCI machine learning repository at different supports. The distributed BitTable Multi-Agent based algorithm has proved to have better performance and execution time.

References

1. Agrawal, R., Imieliński, T., Swami, A.: Mining association rules between sets of items in large databases. *ACM SIGMOD Record* 22(2), 207–216 (1993)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: *Proceedings of the 20th International Conference on Very Large Data Bases*, pp. 487–499. Morgan Kaufmann Publishers Inc., San Francisco (1994)
3. Bodon, F.: A fast apriori implementation. In: *Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations*, vol. 90. Citeseer (2003)
4. Bodon, F.: Surprising results of trie-based FIM algorithms. In: *Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI 2004)*, vol. 126. Citeseer (2004)
5. Bodon, F.: A trie-based APRIORI implementation for mining frequent item sequences. In: *Proceedings of the 1st International Workshop on Open Source Data Mining: Frequent Pattern Mining Implementations*, pp. 56–65. ACM, New York (2005)
6. Bodon, F., Rónyai, L.: Trie: An alternative data structure for data mining algorithms. *Mathematical and Computer Modelling* 38(7-9), 739–751 (2003)
7. Dong, J., Han, M.: BitTableFI: An efficient mining frequent itemsets algorithm. *Knowledge-Based Systems* 20(4), 329–335 (2007)
8. Fakhry, M., Atteya, W.A.: An Enhanced Algorithm for Mining Association Rules. In: *First International Conference on Intelligent Computing and Information Systems* (2002)
9. Park, J., Chen, M., Yu, P.: An effective hash-based algorithm for mining association rules. *ACM SIGMOD Record* 24(2), 175–186 (1995)

10. Park, J., Chen, M., Yu, P.: Using a hash-based method with transaction trimming for mining association rules. *IEEE Transactions on Knowledge and Data Engineering* 9(5), 813–825 (2002)
11. Song, W., Yang, B., Xu, Z.: Index-BitTableFI: An improved algorithm for mining frequent itemsets. *Knowledge-Based Systems* 21(6), 507–513 (2008)
12. Atteya, W.A., Dahal, K., Alamgir Hossain, M.: Multi-agent association rules mining in distributed databases. In: Gaspar-Cunha, A., Takahashi, R., Schaefer, G., Costa, L. (eds.) *Soft Computing in Industrial Applications*. AISC, vol. 96, pp. 305–314. Springer, Heidelberg (2011)
13. Atteya, W.A., Dahal, K., Alamgir Hossain, M.: Multi-agent system for early prediction of urinary bladder inflammation disease. In: 10th International Conference on Intelligent Systems Design and Applications (ISDA 2010), pp. 539–544. IEEE, Los Alamitos (2010)
14. Zaki, M.: Scalable algorithms for association mining. *IEEE Transactions on Knowledge and Data Engineering* 12(3), 372–390 (2002)

Sentiment Analysis of Customer Reviews: Balanced versus Unbalanced Datasets

Nicola Burns, Yaxin Bi, Hui Wang, and Terry Anderson

University of Ulster
burns-n4@email.ulster.ac.uk,
{y.bi,h.wang,tj.anderson}@ulster.ac.uk

Abstract. More people are buying products online and expressing their opinions on these products through online reviews. Sentiment analysis can be used to extract valuable information from reviews, and the results can benefit both consumers and manufacturers. This research shows a study which compares two well known machine learning algorithms namely, dynamic language model and naïve Bayes classifier. Experiments have been carried out to determine the consistency of results when the datasets are of different sizes and also the effect of a balanced or unbalanced dataset. The experimental results indicate that both the algorithms over a realistic unbalanced dataset can achieve better results than the balanced datasets commonly used in research.

Keywords: Sentiment analysis, opinion mining, naïve Bayes, language model.

1 Introduction

In the UK, online retail sales account for over 10% of purchases, and their growth rate is markedly outstripping store-based sales [13]. Many customers express their opinions about products through online reviews. These reviews are key for marketing intelligence since they contain valuable information. Popular products may have hundreds of reviews making it hard for the customer to find the information they require, and as a result there is a need to automatically classify this data. This can benefit both customers and manufacturers of products. Customers can see what other consumers thought about the products', viewing the products strengths and weaknesses. Manufacturers can then see where their product falls short in order to improve it, and also they can compare their products to other competitive products

Opinion mining and sentiment analysis are relatively new areas of research. Research has evolved from classifying whole documents [15], to classifying each sentence [17], to classifying each separate feature of the product [8][9]. This study compares two well-known machine learning techniques, namely naïve Bayes and language model. Previous studies have used a balanced dataset, however in the product domain it is commonly the case that the ratio of positive and negative reviews is unbalanced, therefore this paper focuses on and investigating the effects of the size and ratio of a dataset. Our system architecture takes customer reviews as input to each of the classifiers and outputs the dataset split into positive and negative reviews.

2 Related Work

Sentiment analysis has been the work of many researchers over the past years. The focus of this research is on polarity sentiment analysis of customer product reviews. Similar research include Liu *et al.* [10], Dave *et al.* [5], Hu and Liu [8][9], Ding and Liu [6] and Ding *et al.* [7], where each use a dataset of reviews containing various products. Our work differs in that our dataset will concentrate on one product rather than a combination of products. A considerable amount of research has been carried out using the Internet Movie Review Database (IMDb) [12] and the Polarity Dataset [11]. These datasets have an even number of positive and negative reviews, however in the product domain it is typical that there are substantially more positive reviews compared to negative reviews. Our work will therefore compare the effects of a balanced and unbalanced dataset.

In this research, two approaches have been chosen for our experiments. Three aspects were considered when choosing these techniques; efficiency, accuracy and suitability. From analysing past research both classifiers provide adequate efficiency and accuracy, though with room for improvement. Also, the implementation of both classifiers can be adapted to the context of this work and our future research. Research on sentiment classification shows that language model and naïve Bayes classifiers are two of the most popular and influential classifiers [3][12].

This research is closely related to that of Conrad and Schilder [4] and Ye *et al.* [16]. Conrad and Schilder investigate opinion mining in legal Weblogs, and they compare both naïve Bayes and language model (LM) classifiers. Ye *et al.* compare support vector machines (SVM), naïve Bayes and LM classifiers to investigate tourist reviews. Conrad and Schilder aim to identify the extent and usefulness of opinion mining in the legal domain. Ye *et al.* wish to present sophisticated techniques which have not yet been tailored to the travel domain. Our work investigates using only one product or one product domain to boost classification accuracy and shows that using a realistic unbalanced dataset can perform better than a balanced dataset.

3 Evaluation Procedure

For our experiments, we use a dynamic language model classifier and a naïve Bayes classifier. Classifiers were created using LingPipe, which is a Java-based natural language processing toolkit distributed with source code by Alias-I [1]. Our system architecture is shown in figure 1. Amazon.com provides APIs containing information on products, including customer reviews for each product. The reviews are extracted and then used as training and testing data. This data is used by each of the classifiers to generate results for analysis. The download was performed in September 2010 and the dataset produced consists of reviews of television sets. Customers rate their review on a 5 star scale. For this study, we consider reviews with 4 – 5 stars as positive and 1-2 stars as negative 3 stars are regarded as neutral and therefore ignored. After download the dataset contained 12,374 positive reviews and 2,076 negative reviews. We also compare our results with datasets taken from [2], focusing on particular product domains, namely cameras and kitchens.

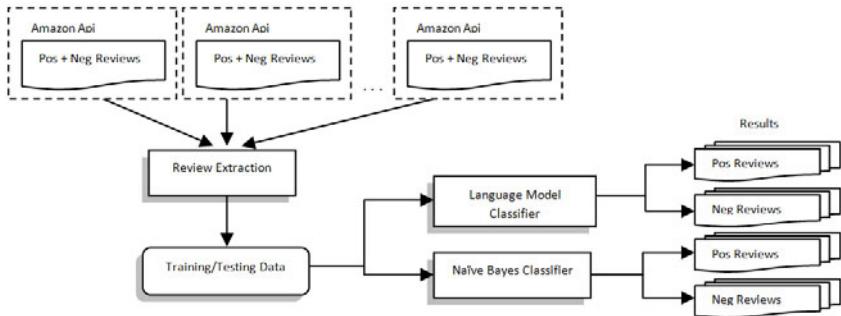


Fig. 1. System Architecture

3.1 Dynamic Language Model

Our model uses an n -gram character-based language model with a generalised form of Witten-Bell smoothing. It uses characters as the basic unit in the algorithm. It provides a probability distribution $p(\sigma)$ over strings $\sigma \in \Sigma^*$ drawn from a fixed alphabet of characters Σ^* . The chain rule: $p(\sigma c) = p(\sigma) \cdot p(c|\sigma)$ factors the joint probability of the string σ along with the character c . Due to the n -gram Markov assumption the context of a conditional estimate $p(c|\sigma)$ is restricted to the last $n - 1$ characters of σ , taking $p(c_n|\sigma_{c_1 \dots c_{n-1}}) = p(c_n|c_1 \dots c_{n-1})$. The maximum likelihood estimator for the model ($p(c_n|c_1 \dots c_{n-1})$) is

$$\hat{p}_{ml}(c|\sigma) = \frac{\text{count}(\sigma c)}{\text{extCount}(\sigma)} \quad (1)$$

where $\text{count}(\sigma c)$ will be the amount of times the string σc appears within the dataset and $\text{extCount}(\sigma) = \sum c \text{ count}(\sigma c)$ will then be the quantity of single character extensions of σ .

3.2 Naïve Bayes

Our model is a traditional token-based approach. It uses a "bag of words" representation, this is a count of tokens occurring in a piece of text. The implementation uses joint probabilities, this means that the naïve Bayes classifier takes a character sequence and returns joint probability estimates of categories and tokens. According to naïve Bayes Rule;

$$p(cat|tokens) = \frac{p(tokens|cat)p(cat)}{p(tokens)} \quad (2)$$

Naive Bayes estimates a multinomial distribution over categories, $p(cat)$. For each category cat , naive Bayes estimates a multinomial distribution over words, which we write as $p(token|cat)$, indicating the dependence of the probability of the *token* on category *cat*.

3.3 Experiments

In order to compare the two classifiers, naive Bayes and language model, we looked at the results based on a balanced and an unbalanced dataset and also the consistency of results when the dataset was different sizes. To conduct our experiments we created the following datasets;

- Unbalanced dataset - all reviews extracted, a realistic representation of the ratio of positive and negative reviews
- Balanced dataset - all negative reviews and the same number of positive reviews.

These datasets are re-sampled at various percentages, e.g. 100%, 90%. We use 8 datasets for unbalanced and 8 for balanced. Table 1 and 2 indicates the number of positive and negative reviews in both datasets. Experiments used a 10-fold cross validation. Each dataset was randomly split into 10 folds, 9 folds used for training and 1 fold used for testing. The average of the 10-folds was then used for performance analysis.

Table 1. Number of reviews in unbalanced dataset

% of data-set	Number of reviews			Camera			Kitchen		
	TV			Camera			Kitchen		
	Neg.	Pos.	Total	Neg.	Pos.	Total	Neg.	Pos.	Total
30%	622	3712	4335	330	1893	2223	1236	4721	5957
40%	830	4950	5780	440	2524	2964	1648	6295	7943
50%	1038	6187	7225	550	3155	3705	2060	7869	9929
60%	1245	7425	8670	659	3785	4444	2471	9442	11913
70%	1453	8662	10115	769	4416	5185	2883	11016	13899
80%	1661	9899	11560	879	5047	5926	3295	12590	15885
90%	1868	11137	13005	989	5678	6667	3707	14163	17870
100%	2076	12374	14450	1099	6309	7408	4119	15737	19856

Table 2. Number of reviews in balanced dataset

% of data-set	Number of reviews			Camera			Kitchen		
	TV			Camera			Kitchen		
	Neg.	Pos.	Total	Neg.	Pos.	Total	Neg.	Pos.	Total
30%	622	622	1244	330	330	660	1236	1236	2472
40%	830	830	1660	440	440	880	1648	1648	3296
50%	1038	1038	2076	550	550	1100	2060	2060	4120
60%	1245	1245	2490	659	659	1318	2471	2471	4942
70%	1453	1453	2906	769	769	1538	2883	2883	5766
80%	1661	1661	3322	879	879	1758	3295	3295	6590
90%	1868	1868	3736	989	989	1978	3707	3707	7414
100%	2076	2076	4152	1099	1099	2198	4119	4119	8238

4 Results

4.1 Balanced Dataset

We first focus on the commonly used balanced dataset. Table 3 and figure 2 show the average 10-fold accuracies of each classifier. Naïve Bayes performs best overall with the TV and Camera datasets. For the Kitchen dataset, the language model performs slightly better. With the balanced dataset results varied as the size of the dataset increased with accuracy ranges between 83.21-91.45%. In [4], a comparison of classifiers was carried out on legal blogs which correspond to our results that the naïve Bayes performs better than the language model. The accuracy we achieve in our experiments on customer reviews is much higher than the results using legal blogs.

Table 3. Accuracies of balanced dataset

% data-set	% Accuracy			Kitchen			Camera		
	TV	Kitchen	Camera	NB	LM	P	NB	LM	P
30%	90.92	88.18	0.0419	84.39	83.21	0.1224	88.79	89.85	0.3816
40%	91.45	89.46	0.0731	85.86	85.22	0.3429	86.70	88.86	0.1240
50%	90.89	89.98	0.3386	86.58	85.63	0.1435	87.36	87.09	0.8154
60%	90.32	89.68	0.3024	86.46	86.69	0.5655	87.02	86.26	0.4966
70%	89.88	89.68	0.7059	86.77	86.94	0.6963	88.29	86.93	0.1131
80%	90.40	90.03	0.4586	86.97	87.33	0.4148	88.22	86.46	0.1344
90%	90.52	89.96	0.3395	86.74	87.17	0.2699	87.87	85.08	0.0052*
100%	90.17	89.83	0.4752	86.72	86.83	0.7414	87.62	86.40	0.1545

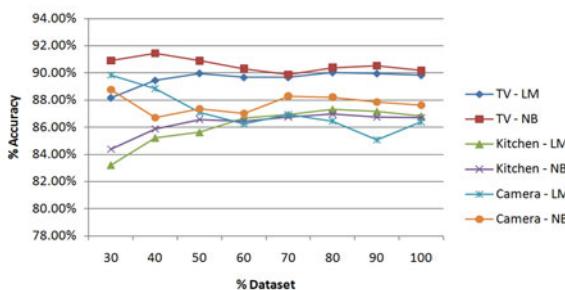


Fig. 2. Accuracies of balanced dataset

4.2 Unbalanced Dataset

Next, we focus on our realistic unbalanced datasets. Table 4 and figure 3 show the accuracy results of our unbalanced dataset. The results show that naïve Bayes outperforms the language model classifier for both TV and Kitchen, but for the Camera dataset the language model performs better. The accuracy of each classifier tends to increase with the size of the dataset.

The statistical difference between the two classifiers varied depended on the data-set. The difference of accuracies between the two classifiers was statistically significant ($p < 0.01$) when the TV dataset was as low as 30% and above 70%. For the Kitchen dataset the classifiers are similar in performance, whereas with the Camera dataset the difference is almost always statistically significant. Ye *et al.* [16] investigate the size of the training set and find the difference between algorithms was extremely significant when the training set was small. Using unbalanced datasets we can achieve accuracies ranging from 87.33-93.83%, which is again better than the results achieved in [4] using legal blogs.

Table 4. Accuracies of unbalanced dataset

% data-set	% Accuracy			Kitchen			Camera		
	TV			NB	LM	P	NB	LM	P
30%	90.84	89.62	0.0006*	88.42	87.33	0.0235	91.41	93.21	0.0179
40%	91.31	90.59	0.0786	88.59	87.35	0.0130	91.60	93.45	0.0024*
50%	91.32	90.48	0.0287	88.50	87.75	0.0417	91.55	93.82	0.0001*
60%	91.18	90.39	0.0143	88.80	88.23	0.0498	91.31	93.25	0.0002*
70%	91.37	90.33	0.0058*	88.95	88.40	0.0549	90.84	92.71	0.0011*
80%	91.39	90.55	0.0032*	89.12	88.74	0.1536	90.62	92.39	0.0002*
90%	91.61	90.77	0.0010*	88.95	88.74	0.1859	90.81	92.46	0.0000*
100%	91.89	91.18	0.0014*	88.98	88.81	0.2136	91.00	92.95	0.0002*

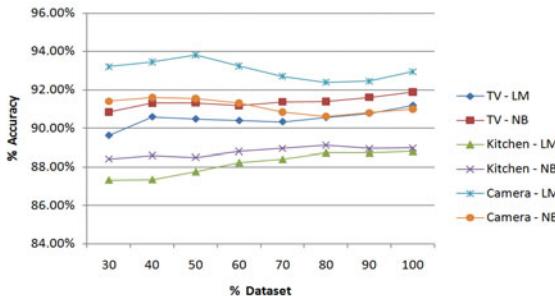


Fig. 3. Accuracies of unbalanced dataset

4.3 Sampling

To make a fair comparison of the two datasets we re-sampled the larger unbalanced dataset to match the size of the smaller balanced dataset, keeping the percentage of positive and negative reviews to scale. The accuracies of the re-sampled unbalanced dataset are shown in figure 4. When comparing the balanced (figure 2) and unbalanced datasets (figure 4) it can be seen that both classifiers generally have greater accuracy with the unbalanced dataset. With the unbalanced dataset there is a steady increase with the size of the dataset, whereas with the balanced dataset the results are somewhat more varied. The Kitchen dataset shows a steady increase with size and the Camera dataset has a decrease with size. Ye *et al.* [16] use a relatively balanced

dataset for their experiments and show an increase in accuracy with an increase in the amount of training data. Therefore, we show that for a realistic unbalanced dataset the accuracy will increase with the size of the dataset and for a balanced dataset the accuracy may decrease depending on the domain.

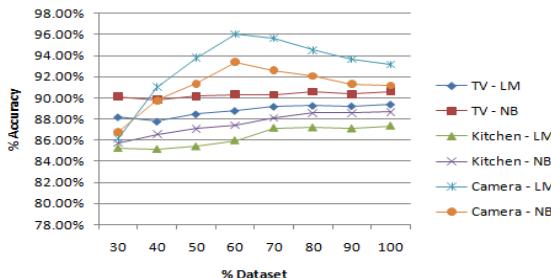


Fig. 4. Accuracies of re-sampled unbalanced dataset

4.4 Precision and Recall Results

Precision is a measure of presenting only relevant terms, the higher the precision the more relevant the results. Figures 5 and 6 show the precision of the balanced and re-sampled unbalanced datasets. Language model performs better on two of three datasets for both the balanced and unbalanced dataset. The unbalanced dataset has precision values ranging from 97.66-99.86%, whereas the balanced dataset ranges from 82.94-93.24%, showing that using an unbalanced dataset can return more relevant results.

When using the unbalanced dataset the precision tends to decrease with the dataset size in all three cases. With the balanced dataset the results varied depending on the dataset. In the Kitchen dataset, the precision increases with size, but in the Camera dataset the precision decreases with size.

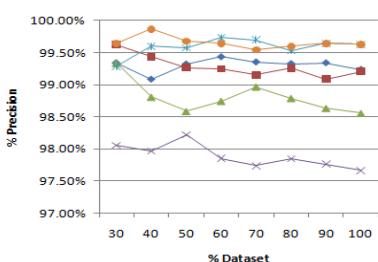


Fig. 5. Unbalanced precision results

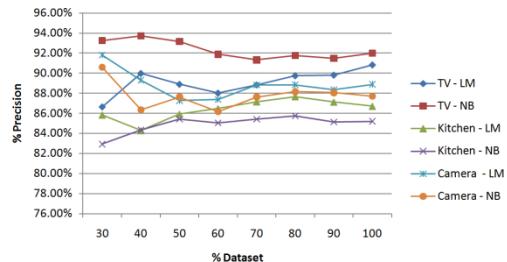


Fig. 6. Balanced precision results

Recall is the ability to retrieve as many relevant terms as possible. Figures 7 and 8 show that for, the language model has better overall results for two of the three datasets using both balanced and unbalanced datasets. The unbalanced dataset has recall values ranging from 87.02-97.94%, whereas the balanced dataset ranges from

49.07-51.45%, showing that using an unbalanced dataset returns additional results but retrieves more relevant results.

As with precision, when using the unbalanced dataset recall normally decreases with the size of the dataset and with the balanced dataset again the results varied considerably depending on the dataset.

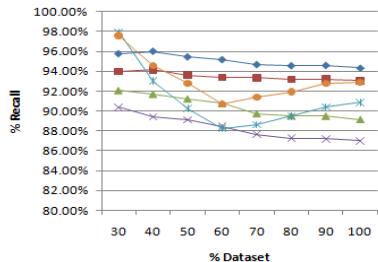


Fig. 7. Unbalanced recall results

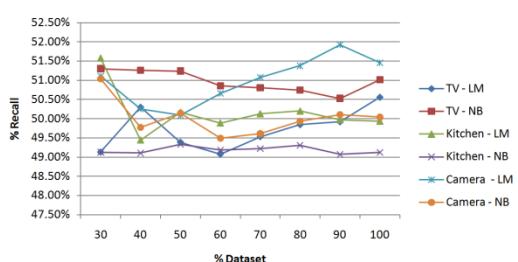


Fig. 8. Balanced recall results

4.5 Comparison of Datasets

An interesting finding is that while previous research used balanced datasets, our realistic unbalanced dataset achieves substantially higher results. Accuracy and precision are higher and recall is considerably higher. On further analysis of our TV balanced and unbalanced datasets, we concluded from table 5 that the higher result achieved by the unbalanced dataset is due to the larger amount of positive reviews. The negative reviews account for 14.37% of the dataset and achieve under 50% accuracy, whereas the positive reviews account for 85.63% of the dataset and achieve almost 99% accuracy. The accuracy of the positive reviews increases the overall accuracy, which is true for all three datasets. Within a given product domain, where positive reviews often outweigh the negative reviews, we have demonstrated that a realistic unbalanced dataset can achieve higher accuracy and precision.

Table 5. Average accuracies split by positive and negative

Average % Accuracy				
	Unbalanced		Balanced	
	NB	LM	NB	LM
Neg	49.57	44.65	88.34	88.82
Pos	98.99	98.99	92.00	90.85
Overall	91.89	91.18	90.17	89.83

Another observation is that for all TV, Kitchen and Camera datasets, each classifier follows the same pattern for the unbalanced dataset, whereas the results were more variable for the balanced dataset. This shows that when using a balanced dataset, different domains can produce very different results, For example, in the unbalanced

datasets, accuracy increases with the size of dataset whereas with the balanced dataset for Camera it decreases and Kitchen dataset it increases with size.

As mentioned, the size of the dataset can impact the results produced. With the unbalanced dataset, regardless of the domain, results follow the same pattern, i.e. they get better as the dataset gets bigger. A conclusion is therefore that the pattern produced is due to the dataset, and when we alter the dataset the pattern is lost. With the balanced dataset the results varied, e.g. with the Camera dataset accuracy is higher when using 30% of the dataset compared with using 100% as opposed to the Kitchen dataset which has higher accuracy at 100%. Therefore, with an unbalanced dataset the bigger the dataset the better the result, but for some unbalanced datasets the results may be better at smaller sizes.

5 Conclusion

This research uses two well-known machine learning techniques: naïve Bayes and the language model to classify customer reviews. We focus on one product type, i.e. TVs, we also compare our results with other datasets focusing on one product domain, i.e. Kitchen and Camera. Our findings indicate that naïve Bayes has much more accurate results using TV reviews, however results are mixed for the Kitchen and Camera datasets. This suggests that the domain can impact the results of the classifier. In our case, the results indicated that naïve Bayes was the preferred classifier with the TV dataset and as it takes a “bag of words” approach, the results suggest that semantics does not seem to play an important role in this domain.

Previous studies in this domain involve using a range of products, our work focused on one product type and for this reason was able to achieve much better results, achieving over 90% accuracy for both classifiers. This may be a result of the data in the training set being only related to the product, therefore when testing is carried out there will be a higher chance of correct classification. If a dataset contains a number of products which may be of a similar category this will reduce the amount of information related to specific products and therefore may reduce performance.

Finally, while previous research used balanced datasets, we demonstrate that a realistic unbalanced dataset can achieve substantially better results. We also show that with a balanced dataset the results varied but with an unbalanced dataset, regardless of the domain, results follow the same pattern.

References

1. Alias-I 2008, LingPipe 3.9.2, <http://www.alias-i.com/lingpipe> (March 1, 2010)
2. Blitzer, J., Dredze, M., Pereira, F.: Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In: 45th Annual Meeting of the Association for Computational Linguistics, pp. 440–447. ACL, Prague (2007)
3. Carpenter, B.: Scaling High-Order Character Language Models to Gigabytes. In: Workshop on Software, pp. 86–99. Association for Computational Linguistics, Morris-town (2005)

4. Dave, K., Lawrence, S., Pennock, D.: Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: 12th International Conference on World Wide Web, pp. 519–528. ACM, New York (2003)
5. Ding, X., Liu, B.: The Utility of Linguistic Rules in Opinion Mining. In: 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 811–812. ACM, New York (2007)
6. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: International Conference on Web Search and Web Data Mining, pp. 231–240. ACM, New York (2008)
7. Hu, M., Liu, B.: Mining opinion features in customer reviews. In: 19th National Conference on Artificial Intelligence, pp. 755–760. AAAI Press / The MIT Press (2004b)
8. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 168–177. ACM, New York (2004a)
9. Jelinek, F., Merialdo, B., Roukos, S., Strauss, M.: A dynamic language model for speech recognition. In: Workshop on Speech and Natural Language, pp. 293–295. Association for Computational Linguistics, Morristown (1991)
10. Liu, B., Hu, M., Cheng, J.: Opinion Observer: Analyzing and Comparing Opinions on the Web. In: Proceedings of the 14th International Conference on World Wide Web, pp. 342–351. ACM, New York (2005)
11. Pang, B., Lee, L.: A Sentimental Education: Sentiment Analysis using subjectivity summarization based on minimum cuts. In: 42nd Annual Meeting on Association for Computational Linguistics, pp. 271–278. Association for Computational Linguistics, Morristown (2004)
12. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment Classification using Machine Learning Techniques. In: ACL 2002 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 79–86. Association for Computational Linguistics, Morristown (2002)
13. Potter, M.: February 1st-last update, European online retail sales up (2010), <http://uk.reuters.com/article/2010/02/01/uk-europe-retail-online-idUKTRE61000G20100201> (March 1, 2011)
14. Thelen, M., Riloff, E.: A Bootstrapping Method for Learning Semantic Lexicons using Extraction Pattern Contexts. In: ACL 2002 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 214–221. Association for Computational Linguistics, Morristown (2002)
15. Turney, P.D.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In: Proceedings of the 40th Annual Meeting of the ACL, pp. 417–424. Association for Computational Linguistics, Morristown (2002)
16. Ye, Q., Zhang, Z., Law, R.: Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert Systems with Applications: An International Journal 36(3), 6527–6535 (2009)
17. Yu, H., Hatzivassiloglou, V.: Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 129–136. Association for Computational Linguistics, Morristown (2003)

Agents' Logics with Common Knowledge and Uncertainty: Unification Problem, Algorithm for Construction Solutions

Vladimir V. Rybakov

School of Computing, Mathematics and IT,

Manchester Metropolitan University,

Manchester M1 5GD, UK, and

Institutes of Mathematics, Siberian Federal University, Krasonoyarsk, Russia

V.Rybakov@mmu.ac.uk

Abstract. This paper studies agents' logics with operation *uncertainty*. Prime attention is paid to agents' common knowledge logics with logical operation uncertainty and logical unification in these logics. The unification problem is: for two arbitrary given formulas with meta-variables (coefficients) to answer whether they are unifiable, and if yes to construct a unifier. This problem is equivalent to problem of solvability logical equations with coefficients and finding theirs solutions. We show that the basic common knowledge logic with uncertainty operation (notation - $CKL_{n,U}$) is decidable w.r.t. logical unification of the common knowledge formulas, and that for unifiable common knowledge formulas we can construct a unifier (we may describe solving algorithm). This result is extended to a wide class of logics expanding $CKL_{n,U}$.

Keywords: multi-agent systems, agents' logic, common knowledge, uncertainty, unification.

1 Introduction, Background

Technique of AI nowadays broadly uses instruments based on intelligent agents. Agents, as entities, might be understood very broadly, but often they are described schematically as an abstract functional systems similar to a computer programs. Sometimes definitions of intelligent agents emphasize their autonomy, and so prefer the term autonomous intelligent agents, at the same time, often, interaction of agents, cooperation, is accepted as very desirable feature. Areas of applications are utterly diverse, but, anyway they are primarily focused to IT (cf. Nguyen et al. [18][19][20]). These wide areas evidently need technique and models to reason about agents' knowledge and properties. In particular, technique of symbolic logic is widely used (cf. [12][13][14]). Nowadays research in computer science oriented to knowledge representation actively uses various languages and logical systems to capture elements of human reasoning and computational aspects. These logical systems provide us with various inference capabilities to deduce implicit knowledge from the explicitly represented knowledge,

as well as with explicit, mathematically precise, description of properties of the objects. Knowledge representation structures actively involve logical language, cf. Brachman and Schmolze (1985, [6]), Moses and Shoham (1993, [15]), Nebel (1990, [16]), Quantz and Schmits (1994, [21]), and has applications in industry Rychtycki (1996, [29]). Research of agent-decision oriented systems paid many attention to formal descriptions of the meaning of agents knowledge.

Because often logical systems are assumed to be multi-agents, the question what is a shared knowledge and what is a common knowledge for all agents has been risen. Some initial concepts concerning this subject can be found in Barwise (1988, [7]), Niegerand and Tuttle (1993, [17]), Dvorek and Moses (1990, [8]). Well developed approach to common knowledge logics is contained in Fagin R., Halpern J., Moses Y., Vardi M. (1995, [12]). In particular, there is a series of theorems on completeness for various common knowledge logics w.r.t. possible worlds models. Many aspects of this theory are in study from distinct points of view, and in our research we investigate open problem about unification in common knowledge logics.

One important feature within multi-agent approach is possible *uncertainty* of initial information and agents' knowledge. In computer science, uncertain data is the notion of data that contains specific uncertainty. When representing such data in a database, some indication of the probability of the various values is used to cope with the value of data. Often to analyze how data are certain, elements of probability theory are involved: a set of possible states or outcomes where probabilities are assigned to each possible state or outcome is taken for further inference and study. We would like to analyze logical properties of uncertainty by introduction special logical operator in the language of multi-agents' logic.

If we intend to undertake logical research, one of general problems arising at once: unification problem. Unification is one of most widely used techniques in Computer Science (especially term rewriting), it forms a core of such Declarative Programming Languages as Prolog. Unification is an operation in computer science and logic which produces from two logic terms a substitution which either identifies the terms (in the case of syntactic unification) or makes the terms equal modulo some equational theory (in the case of semantic unification). Unification technique is widely used in automated reasoning, logic programming and programming language type system implementation.

This technique works effective, though generalized logical unification in first-order logics is, as a rule, undecidable, cf. for instance, Rybakov (1999, [23]). To say more, logical unification, in general, helps to understand, what is the real logical content of the properties described by formulas, whether they can have the same logical context. In particular, a robust research of logical unification in Description Logics is presented in Franz Baader and Silvio Ghilardi [5], Baader and Narendran (2001, [1]) Baader and Küsters (2001, [2]). This can be used for representation of terminological knowledge of application domains. Still a point is that unification problems matches well with equational logic and mathematical background of commutative theory, cf. Baader (1993, [3]). In modal and

intuitionistic logics, a strong theory allowing effectively compute *best* unifiers is developed in works of Silvio Chilardi (cf. [9][10][11]).

In our current paper we study solvability of logical unification for formulas in Common Knowledge Logics with Uncertainty. Main focus is the problem whether the unification problem for such logics is decidable, i.e. whether we can resolve, for arbitrary two given formulas with meta-variables (parameters), are they unifiable, and, if yes, to provide an unifier. We prove that the basic common knowledge logic with logical uncertainty operation $CKL_{n,U}$ is decidable w.r.t. logical unification of the common knowledge formulas (with coefficients), and that for unifiable common knowledge formulas we can construct an unifier (to give an algorithm for). In the final part of our paper this result for the logic $CKL_{n,U}$ is extended to a fairly wide class of logics relative to $CKL_{n,U}$. This paper is based on our previous publication (Rybakov, 2002, [24]), and extend it by study logics with operation of logical uncertainty, we also essentially use technique from Rybakov [22][25][26][27][28].

2 Preliminaries

For reader convenience we start from basic definitions and notation. To describe *common knowledge logics*, recall that the language of CKL_n consists of a countable set P of propositional letters, logical operations of PC (classical propositional logic), and a finite tuple of modal-like unary logical operations K_1, \dots, K_n . The definition of well-formed formula (wff, in the sequel) is as usual and $K_i A$ has the informal meaning: the i -th agent knows A . Possible-Worlds models for knowledge offered in [12] are multi-dimensional Kripke models $\mathcal{M} := \langle S, R_1, \dots, R_n, V \rangle$ with n accessibility relations and valuations V for propositional letters, for any propositional letter p , $\forall a \in S, a \Vdash_V p$ means $a \in V(p)$. The valuation V can be extended from propositional letters to formulas constructed by Boolean operations in the standard way, and $a \Vdash_V K_i A \Leftrightarrow_{def} \forall b \in S (a R_i b \Rightarrow b \Vdash_V A)$. The compound *know* operation E_G , for any set G of agents is introduced as *all agents from G know A* : $E_G A := \bigwedge_{\forall i \in G} K_i A$.

To incorporate common knowledge, the language was extended by adjoining unary logical operations C_G for all sets G of agents with the formation rule: if A is a wff then $C_G A$ is a wff again. The informal meaning of $C_G A$ is: A is common knowledge for all agents from G . The valuations of formulas in possible-worlds models \mathcal{M} can be extended to formulas in new enriched language by: for all $c \in \mathcal{M}$, $c \Vdash_V C_G A \Leftrightarrow \forall k \geq 1, (c \Vdash_V E_G^k A)$, where $k \geq 1$, $E_G^1 := E_G$, $E_G^{k+1} := E_G E_G^k$. We say a formula A is valid in \mathcal{M} , and write $\mathcal{M} \Vdash A$, iff for any $c \in \mathcal{M}$, $c \Vdash_V A$. Fur further information concerning CKL_n and its axiomatic system L we refer to [12].

The well known completeness theorem for CKL_n (cf. e.g. [12]) says that, for this axiomatic system L , $\vdash_L A$ iff A is valid in all finite possible world models. A number of similar theorems, with restrictions to the structure of possible worlds models (as transitivity, reflexivity etc.) and corresponding enlarging of axiomatic systems with new axioms responsible for these restrictions, were proved in [12] also.

Basing on this results, for any class \mathcal{K} of possible worlds frames with described above structure (fixed n , and n binary accessibility relations), we introduce common knowledge logic generated by \mathcal{K} as $CKL(\mathcal{K}) := \{A \mid \forall \mathcal{M} \in \mathcal{K} (\mathcal{M} \Vdash A)\}$. In the light of completeness theorems mentioned above, if \mathcal{K} consists of all possible worlds models with n accessibility relations, $CKL(\mathcal{K})$ is the smallest common knowledge logic CKL_n . Common knowledge logics are all logics generated by appropriate classes of possible-worlds models. For any common knowledge logic L and a formula A , $L \vdash A$ is another notation for $A \in L$.

To incorporate uncertainty, we suggest a new unary logical operator U , and extend formation rules for formulas by: if A is a formula UA is a formula as well. This operation may be nested, so, say $U\neg(A \rightarrow \neg UA)$ is a formula.

The meaning of this operation in semantic context is as follows. If our logic has n -agents and C_n is the corresponding common knowledge operation, then we suggest, for any formula A ,

$$UA := \neg C_n A \wedge \neg C_n \neg A.$$

So, A is uncertain if it is common knowledge that A may be true and may be false. Besides, we can use *local* uncertainty operation U_l , defining it as

$$U_l A := [\bigvee_{1 \leq i \leq n} \neg K_i A] \wedge [\bigvee_{1 \leq i \leq n} \neg K_i \neg A].$$

The mining for local uncertainty would be A is locally uncertain if some agents know that A is true, and some know that A is false.

Because all suggested operations for uncertainty, as we see, are explicitly defined within standard language of CKL_n , the resulting logic with general and local uncertainty will be decidable, in particular, satisfiability problem for this logic is decidable, which follows directly from mentioned above old decidability results for CKL_n itself. The situation with unification decidability is much more difficult, we cannot directly use known up to now results, and we will study this problem in the rest of our paper. For the rest of the paper we denote by $CKL_{n,U}$ minimal common knowledge logic with uncertainty operation U .

3 Logical Unification, Decidability Algorithm

First, we recall basic definitions concerning unification. For any two formulas A and B , $A \equiv B$ is the abbreviation for the formula $(A \rightarrow B) \wedge (B \rightarrow A)$. Let $A(x_1, \dots, x_n, p_1, \dots, p_m)$ and $B(x_1, \dots, x_n, p_1, \dots, p_m)$ be two formulas in the language of a propositional logic L constructed out of certain collection of letters for variables x_1, \dots, x_n and propositional letters p_1, \dots, p_m by means of logical operations from L . We say these formulas are unifiable in L , and write

$$A(x_1, \dots, x_n, p_1, \dots, p_m) \approx_{U,L} B(x_1, \dots, x_n, p_1, \dots, p_m)$$

iff there are formulas C_1, \dots, C_n in the language of L such that

$$L \vdash A(C_1, \dots, C_n, p_1, \dots, p_m) \equiv B(C_1, \dots, C_n, p_1, \dots, p_m).$$

The tuple of formulas C_1, \dots, C_n is said to be an unifier for formulas A and B .

It is clear that if an unifier exists then there is an unifier which consists of formulas built up on propositional letters p_1, \dots, p_n only as L is closed w.r.t. substitutions. Therefore we call p_1, \dots, p_n meta-variables. We say that the *unification problem for a logic L is decidable*, or L is decidable w.r.t. unification, iff there is an algorithm which, for any two given formulas answers whether they are unifiable in L .

We start from study unification in common knowledge logic $CKL_{n,U}$. Initially we restrict our attention to only common knowledge formulas - formulas which use only common knowledge operation except Boolean operations.

Note that these formulas include uncertainty operation in whole volume because this operation is expressible in terms of common knowledge operation. Actually we will follow close to material from our paper [24], taking care on the results would remain true for our extended case of usage logical uncertainty operation. For any common knowledge logic L with n -agents, we say the unification problem for common knowledge formulas is solvable iff there is an algorithm recognizing pairs of unifiable formulas, which are constructed from variables and meta-variables by only Boolean operations and the common knowledge operation C_n , where n is the number of all agents for L . We will call such formulas *common knowledge formulas*.

For any common knowledge logic L , M_L is the set of all common knowledge formulas A such that $L \vdash A$ holds. It is easy to see that for any given common knowledge logic L with n agents,

$$L \vdash C_n(p \rightarrow q) \rightarrow (C_n p \rightarrow C_n q), \quad (1)$$

$$L \vdash C_n p \rightarrow C_n C_n p. \quad (2)$$

Also, for any L , M_L is closed w.r.t. the rule $x/C_n x$ as well as modus ponens and substitutions of M_L -like formulas. Therefore any M_L is a modal logic extending the smallest normal transitive modal logic $K4$. In what follows we will call M_L the *wrapping modal logic* of the common knowledge logic L . Exactly as in [24] we can derive

Lemma 1. *For common knowledge logic with uncertainty U ,*

$$M_{CKL_{n,U}} = K4,$$

i.e. $M_{CKL_{n,U}}$ coincides with the modal logic $K4$.

Now we have to show that the facts of unifiability in the original logic $CKL_{n,U}$ and in its wrapping modal logic $M_{CKL_{n,U}}$ coincide.

Lemma 2. *If $CKL_{n,U}$ is the smallest common knowledge logic with n -agents and uncertainty operation U , then for any two common knowledge formulas A and B with meta-variables,*

$$[A \approx_{U, CKL_{n,U}} B] \Leftrightarrow [A \approx_{U, M_{CKL_{n,U}}} B].$$

Thus, applying Lemmas 1, 2 and Theorem 6.1.23 from [22], which says that admissibility of inference rules with meta-variables in $K4$ is decidable, and that for rules which are not admissible we can effectively construct an obstacle, we immediately derive

Theorem 1. *Unification problem in common knowledge logic with uncertainty $CKL_{n,U}$ for common knowledge formulas is decidable: there is an algorithm verifying for any two common knowledge formulas A and B if they are unifiable in $CKL_{n,U}$ and constructing a unifier if yes.*

Based on this result, we would like to extend it to common knowledge logics L expanding $CKL_{n,U}$, stronger than $CKL_{n,U}$. First, we have to collect some statements concerning wrapping modal logics.

Let L be a Common Knowledge Logic which has n -agents in the language, and let $\mathcal{M} := \langle S, R_1, \dots, R_n \rangle$ be a L -frame, i.e. $\forall A \in L, \mathcal{M} \Vdash A$, that is $\mathcal{M} \in Mod(L)$. Let $Cl(\mathcal{M}) := \langle S, R^* \rangle$, where

$$\forall a, b \in S, aR^*b \Leftrightarrow [\exists R_j(aR_jb)] \vee$$

$$\vee [\exists a_1, \dots, a_m \in S (a_1 = a \& a_m = b \& (\forall i (1 \leq i < m) \exists R_j(a_i R_j a_{i+1}))).]$$

We will call $Cl(\mathcal{M})$ the total close of the frame \mathcal{M} . Let

$$Mod_{Cl}(L) := \{Cl(\mathcal{M}) \mid \mathcal{M} \in Mod(L)\},$$

i.e. $Mod_{Cl}(L)$ consists of all total closes of all L -frames. Also for any set \mathcal{Z} of n -dimensional possible worlds frames, $Cl(\mathcal{Z}) := \{Cl(\mathcal{M}) \mid \mathcal{M} \in \mathcal{Z}\}$. For any set S of multi-dimensional Kripke frames, S_{Fin} is the set of all finite frames from S .

Lemma 3. *For any set \mathcal{Z} of n -dimensional possible worlds frames, $Cl(\mathcal{Z}) \subseteq Mod(K4)$.*

Lemma 4. *For any common knowledge logic L extending $CKL_{n,U}$, $Mod(K4)_{Fin} \subseteq Mod_{Cl}(L) \implies (M_L = K4)$.*

We say a common knowledge logic L has the finite model property if there is a class \mathcal{K} , may be infinite, but consisting of only finite possible worlds frames, such that $L = L(\mathcal{K}) := \{A \mid \forall \mathcal{M} \in \mathcal{K}, \mathcal{M} \Vdash A\}$.

Lemma 5. *For any common knowledge logic L extending $CKL_{n,U}$ if the following $Mod(K4)_{Fin} \subseteq Mod_{Cl}(L)$ holds and if L has the finite model property then, for any two common knowledge formulas A and B with meta-variables, $[A \approx_{U,L} B] \Leftrightarrow [A \approx_{U,K4} B]$.*

Now applying Lemmas 4, 5 and Theorem 6.1.23 from [22] about decidability of admissibility for rules with meta-variables (coefficients) in specific modal logics we immediately obtain

Theorem 2. Let L be a common knowledge logic L extending $CKL_{n,U}$ and let $Mod(K4)_{Fin} \subseteq Mod_{Cl(L)}$ and let L to have the finite model property. Then the unification problem in L for common knowledge formulas is decidable and there is an algorithm constructing unifiers.

To extend obtained results more, for the rest of this paper we will consider only common knowledge logics L with n -agents which possess the finite model property, i.e. $L = L(\mathcal{K})$ where \mathcal{K} is a set of finite possible worlds frames. We will directly use terminology and notation concerning constructible k -characterizing models $Ch_k(M_L)$ for the wrapping modal logics M_L , for all $k \in N$ (cf. [22]). Since now we follow the paper (Rybakov, [24]), extending results on our logics with uncertainty operation.

Definition 1. For a given common knowledge logic L , if there is a finite frame $\mathcal{F} \in Mod_{Cl}(L)$ rooted by an irreflexive element of depth $u > m$ and \mathcal{F} has d immediate successor clusters then we say L admits irreflexive d -branching below m . If there is a finite rooted frame $\mathcal{F} \in Mod_{Cl}(L)$ with the root consisting of k -element reflexive cluster C which has depth strictly more than m and has d immediate successor clusters then we say that L admits reflexive k, d -branching below m .

Definition 2. We say that L has the generalized property of branching below m if the following hold:

- (i) If L admits the irreflexive d -branching below m , and \mathcal{F} is a finite generated subframe, with depth which is not less than m , of a component of $Ch_k(M_L)$, where M_L is the wrapping modal logic for L and \mathcal{F} has n roots, where $n \leq d$, then the frame $\mathcal{F}_1 := \mathcal{F} \oplus 1_{IR}$ (i.e. \mathcal{F}_1 is obtained from \mathcal{F} by adding a new irreflexive one-element root 1_{IR}) belongs to $Mod_{Cl}(L)$.
- (ii) If L admits the reflexive k, d -branching below m then the following holds. If \mathcal{F} is a finite generated subframe of a component of $Ch_k(M_L)$, which has depth not less than m and has n roots, where $n \leq d$, then the frame $\mathcal{F}_1 := \mathcal{F} \oplus C$, which is obtained from \mathcal{F} by adding the root consisting of the reflexive cluster C with k elements, belongs to $Mod_{Cl}(L)$.

Definition 3. We say that a common knowledge logic L has the effective extension property if there is a computable function $g(x)$ such that the following hold. For every finite L -frame \mathcal{F} and arbitrary subframe \mathcal{F}_1 of \mathcal{F} there is a L -subframe \mathcal{F}_2 of \mathcal{F} with $\mathcal{F}_1 \preceq \mathcal{F}_2$, i.e. \mathcal{F}_1 is a subframe of \mathcal{F}_2 , where $|||\mathcal{F}_2||| \leq g(|||\mathcal{F}_1|||)$ and $|||\mathcal{F}_j|||$ is the number of clusters of the base set of \mathcal{F}_j .

Lemma 6. For any common knowledge logic L extending CKL_n , if L has

- (i) The finite model property,
- (ii) The generalized branching property below m , for some fixed m ,
- (iii) The effective extension property, then the following holds. For any two common knowledge formulas A and B with meta-variables,

$$[A \approx_{U,L} B] \Leftrightarrow [A \approx_{U,M_L} B].$$

Involving Lemma 6 and some revised version of Theorem 6.1.23 from [22] on decidability of admissibility for rules with meta-variables and finding unifiers in modal logics we obtain

Theorem 3. *If L is a common knowledge logic extending $CKL_{n,U}$ and L has*

- (i) *The finite model property,*
 - (ii) *The generalized branching property below m , for some fixed m ,*
 - (iii) *The effective extension property,*
- then the unification problem in L for common knowledge formulas is decidable and there is an algorithm constructing unifiers.*

4 Conclusion

The paper develops a technique to construct mathematical models for agents' common knowledge logics with logical operation uncertainty, which adequately describe logical uncertainty via agents' operations. Prime aim of the paper is study of the unification problem for formulas with coefficients in such logics. In this paper we prove that the basic common knowledge logic with uncertainty operation (notation - $CKL_{n,U}$) is decidable w.r.t. logical unification of the common knowledge formulas. Also we find an algorithm which constructs a unifier for unifiable formulas. In the final part of this paper we extended this result to a wide class of logics expanding $CKL_{n,U}$.

Obtained results have primarily theoretical value, though they may be applied for describing general concepts in agents' logic, as well as for determination possibility to express specifications of a given sorts (in terms of agents' logic, written as formulas) via given pre-specified specs.

References

1. Baader, F., Narendran, P.: Unification of Concept Terms in Description Logics. *J. Symbolic Computation* 31(3), 257–305 (2001)
2. Baader, F., Küsters, R.: Unification in Description Logics with Transitive Closure of Roles. In: Nieuwenhuis, R., Voronkov, A. (eds.) LPAR 2001. LNCS (LNAI), vol. 2250, pp. 217–232. Springer, Heidelberg (2001)
3. Baader, F., Snyder, W.: Unification Theory. In: Handbook of Automated Reasoning, pp. 445–532 (2001)
4. Baader, F., Morawska, B.: Unification in the Description Logic EL. In: Treinen, R. (ed.) RTA 2009. LNCS, vol. 5595, pp. 350–364. Springer, Heidelberg (2009)
5. Baader, F., Ghilardi, S.: Unification in modal and description logics. *Logic J. of IGPL* (2010); First published online (April 29, 2010), doi:10.1093/jigpal/jzq008
6. Brachman, R.J., Schmolze, J.G.: An overview on the KL-ONE knowledge representation system. *Cognitive Science* 9(2), 179–226 (1985)
7. Barwise, J.: Three Views of Common Knowledge. In: Vardi (ed.) Proc. Second Conference on Theoretical Aspects of Reasoning about Knowledge, pp. 365–379. Morgan Kaufmann, San Francisco (1988)
8. Dwork, C., Moses, Y.: Knowledge and Common Knowledge in a Byzantine Environment: Crash Failures. *Information and Computation* 68(2), 156–183 (1990)

9. Ghilardi, S.: Unification, finite duality and projectivity in varieties of Heyting algebras. *Ann. Pure Appl. Logic* 127(1-3), 99–115 (2004)
10. Ghilardi, S., Sacchetti, L.: Filtering unification and most general unifiers in modal logic. *J. Symb. Log.* 69(3), 879–906 (2004)
11. Ghilardi, S.: Best Solving Modal Equations. *Ann. Pure Appl. Logic* 102(3), 183–198 (2000)
12. Fagin, R., Halpern, J., Moses, Y., Vardi, M.: Reasoning About Knowledge. The MNT Press, Cambridge (1995)
13. Kifer, M., Lozinski, L.: A Logic for Reasoning with Inconsistency. *J. Automated Deduction* 9, 171–215 (1992)
14. Kraus, S., Lehmann, D.L.: Knowledge, Belief, and Time. *Theoretical Computer Science* 98, 143–174 (1988)
15. Moses, Y., Shoham, Y.: Belief and Defeasible Knowledge. *Artificial Intelligence* 64(2), 299–322 (1993)
16. Nebel, B.: Reasoning and Revision in Hybrid Representation Systems. LNCS, vol. 422. Springer, Heidelberg (1990)
17. Neiger, G., Tuttle, M.R.: Common knowledge and consistent simultaneous coordination. *Distributed Computing* 5(3), 334–352 (1993)
18. Nguyen, N.T., et al. (eds.): KES-AMSTA 2008. LNCS (LNAI), vol. 4953. Springer, Heidelberg (2008)
19. Nguyen, N.T., Huang, D.S.: Knowledge Management for Autonomous Systems and Computational Intelligence. *Journal of Universal Computer Science* 15(4) (2008)
20. Nguyen, N.T., Katarzyniak, R.: Actions and Social Interactions in Multi-agent Systems. Special issue for *International Journal of Knowledge and Information Systems* 18(2) (2009)
21. Quantz, J., Schmitz, B.: Knowledge-based disambiguation of machine translation. *Minds and Machines* 9, 97–99 (1996)
22. Rybakov, V.V.: Admissible Logical Inference Rules. *Studies in Logic and the Foundations of Mathematics*, vol. 136. Elsevier Sci. Publ., North-Holland, New-York, Amsterdam (1997)
23. Rybakov, V.V.: Logics of Schemes and Admissible Rules for First-Order Theories. *Stud. Fuzziness Soft. Comp.* 24, 566–879 (1999)
24. Rybakov, V.V.: Unification in Common Knowledge Logics. *Bulletin of the Section Logic* 3(4), 207–215 (2002)
25. Babenyshev, S., Rybakov, V.: Describing Evolutions of Multi-Agent Systems. In: Velásquez, J.D., Ríos, S.A., Howlett, R.J., Jain, L.C. (eds.) KES 2009. LNCS, vol. 5711, pp. 38–45. Springer, Heidelberg (2009)
26. Rybakov, V.: Linear Temporal Logic LTK_K extended by Multi-Agent Logic K_n with Interacting Agents. *J. Log. Comput.* 19(6), 989–1017 (2009)
27. Rybakov, V.: Algorithm for Decision Procedure in Temporal Logic Treating Uncertainty, Plausibility, Knowledge and Interacting Agents. *IJIIT* 6(1), 31–45 (2010)
28. Rybakov, V.: Interpretation of Chance Discovery in Temporal Logic, Admissible Inference Rules. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010. LNCS, vol. 6278, pp. 323–330. Springer, Heidelberg (2010)
29. Rychtyckyi, N.: DLMS: An evaluation of KL-ONE in the automobile industry. In: Aiello, L.C., Doyle, J., Shapiro, S. (eds.) Proc. of the 5-th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 1996), Cambridge, Mass, pp. 588–596. Morgan Kaufmann, San Francisco (1996)

Labelled Transition System Generation from Alvis Language^{*,**}

Leszek Kotulski, Marcin Szpyrka, and Adam Sedziwy

AGH University of Science and Technology,
Department of Automatics, al. Mickiewicza 30, 30-059 Krakow, Poland
`{kotulski,mszpyrka,sedziwy}@agh.edu.pl`

Abstract. Alvis is a modelling language designed for the modelling and formal verification of embedded systems. The key concept of Alvis is an *agent* that denotes any distinguished part of a considered system with defined identity persisting in time. Alvis combines a graphical modelling of interconnections among agents with a high level programming language used for describing a behaviour of agents. The basic property of the Alvis Toolkit is the ability of generating of a formal system description directly from the Alvis source code. A way of generating Labelled Transition Systems for Alvis models is presented in the paper.

1 Introduction

Alvis [1], [2] is the novel modelling language designed for real-time systems, especially for embedded ones. The main goal of the Alvis project was to strike a happy medium between formal and practical modelling languages. Formal methods like Petri nets, process algebras or time automata are used in real IT projects very rarely due to their specific mathematical syntax. The Alvis syntax is more user-friendly. From programmers point of view, it is necessary to design two layers of an Alvis model. The code layer uses Alvis statements supported by the Haskell functional programming language to define a behaviour of individual agents. The graphical layer (communication diagrams) is used to define communication channels between agents. The layer takes the form of a hierarchical graph, that allows designers to combine sets of agents into modules that are also represented as agents (called *hierarchical ones*). Alvis modelling environment called *Alvis Toolkit* creates in parallel a model of a considered embedded system and the corresponding LTS graph (Labeled Transition System) that is its formal representation. The LTS graph can be formally verified with the help of the CADP toolbox [3].

The paper is organized as follows. Section 2 provides the short presentation of the Alvis modelling language. The formal definition of an agent state is introduced in Section 3. The generation of an LTS graph for a single agent is

* The paper is supported by the Alvis Project funded from 2009-2010 resources for science as a research project.

** Project web site: <http://fm.ia.agh.edu.pl>

described in Section 4 and the definition of LTS for the whole system in presented in Section 5. The algorithm of merging single LTSs into LTS representing the whole system is presented in Section 6. The computational complexity of this algorithm is also estimated in this section. Finally, an example of LTS merging is presented in Section 7.

2 Alvis at a Glance

Alvis is a successor of the XCCS modelling language [4], [5], which was an extension of the CCS process algebra [6], [7]. However, instead of algebraic equations, Alvis uses a high level programming language based on the Haskell syntax.

An Alvis model consists of three layers, but the last one (*system layer*) is predefined. The system layer is used for the simulation and analysis (generation of an LTS graph) purposes. An Alvis model is a system of agents that usually run concurrently, communicate one with another, compete for shared resources etc. The agents, in Alvis, are used for the design of communication diagrams (see Fig. 2). *Active agents* (agent A) perform some activities and are similar to tasks in the Ada programming language [8], [9]. Each of them can be treated as a thread of control in a concurrent system.

An agent can communicate with other agents through *ports* drawn as circles placed at borders of rounded boxes or rectangles. A *communication channel* is defined for two agents and connects two ports. Communication channels are drawn as lines. One-way communication channel (connection $(X_1.p, X_2.q)$) contain an arrowhead that points out the input port for the particular connection.

The code layer is used to define data types used in a considered model, functions for data manipulation and behavior of individual agents. The layer uses

Table 1. Some Alvis statements

Statement	Description
<code>exec x = e</code>	Evaluates the expression and assign the result to the parameter; the <code>exec</code> keyword can be omitted.
<code>if (g1) {...}</code> <code>elseif (g2) {...}</code> ... <code>else {...}</code>	Conditional statement.
<code>in p</code>	Collects a signal via the port <code>p</code> .
<code>in p x</code>	Collects a value via the port <code>p</code> and assigns it to the parameter <code>x</code> .
<code>loop (g) {...}</code> <code>loop {...}</code>	Repeats execution of the contents while the guard if satisfied. Infinite loop.
<code>out p</code>	Sends a signal via the port <code>p</code> .
<code>out p x</code>	Sends a value of the parameter <code>x</code> via the port <code>p</code> ; a literal value can be used instead of a parameter.
<code>select {</code> <code>alt (g1) {...}</code> <code>alt (g2) {...}</code> ... <code>}</code>	Selects one of the alternative choices.

the Haskell functional language (e.g. the Haskell type system) and original Alvis statements. The set of AlvisCL statements is given in Table 1. To simplify the syntax, following symbols have been used. A stands for an agent name, p stands for a port name, x stands for a parameter, g, g1, g2,... stand for guards (Boolean conditions), e stands for an expression and ms stands for milliseconds. Each non-hierarchical agent placed in the communication diagram must be defined in the code layer and vice-versa.

3 Models

An embedded system designed with the help of an object abstraction (expressed by hierarchical agents) can be finally represented by a set of non-hierarchical agents cooperating in the way described by the maximal flat representation of the communication diagram. The polynomial algorithm of designing such a flat representation was described in [2].

Primarily we are interested in a characteristic of states of active agents. A current state of an agent is represented by a tuple with four pieces of information: agent mode (am), program counter (pc), context information list (ci), parameters values tuple (pv).

An active agent can be in one of the following modes: *finished*, *init*, *running*, *waiting*. An Alvis model contains a fixed number of agents. In other words, there is no possibility to create or destroy agents dynamically. If an active agent starts in the *init* mode, it is inactive until another agent activates it with the *start* statement. Active agents that are initially activated are distinguished in the communication diagram – their names are underlined. If an agent is in the *init* mode, its pc is equal to zero and ci is empty.

The *finished* mode means that an agent has finished its work or it has been terminated using the *exit* statement. The statement is argumentless and an agent can terminate its work itself only. If an agent is in the *finished* mode, its pc is equal to zero and ci is empty. The *waiting* mode means that an active agent is waiting for a synchronous communication with another active agent. In such a case, the pc points out the index of the current statement and ci contains the information determining the cause of awaiting and a condition for resuming its work. The last mode, *running*, means that an agent is performing one of its statements. The pc points out an index of the next agent statement.

The formal definition of an agent state is as follows.

Definition 1. A state of an agent X is a tuple $S(X) = (am(X), pc(X), ci(X), pv(X))$, where $am(X)$, $pc(X)$, $ci(X)$ and $pv(X)$ denote mode, program counter, context information list and parameters values of the agent X respectively.

4 Labelled Transition System for Single Active Agent

A state of an agent can be changed as a the result of executing a step. Let us focus on the step idea. Statements such as *exec*, *exit*, *in*, *jump*, *null*, *out*

and *start* are *single-step* statements. On the other hand, *if*, *loop* and *select* are *multi-step* statements. We use recursion to count the number of steps for multi-step statements. For each of these statements, the first step enters the statement body. Then, we count following steps inside curly brackets.

```

agent A {
    i :: Int = 0;
    x :: Int = 1;
    loop (x <> 0) {                                -- 1
        select {                                     -- 2
            alt (i == 0 && ready [in(p)]) {
                in p x; i = 1; }                      -- 3, 4
            alt (i == 1 && ready [out(q)]) {
                out q(x); i = 0; }{                   -- 5, 6
                if(i == 1) { out p; }                  -- 7, 8
                else { null; } } }                     -- 9
            exit;                                    -- 10
    }
}

```

Listing 1.1. Steps counting in Alvis code

Let us consider the piece of code shown in Listing 1.1. It contains 9 steps. Steps numbers are put inside comments. For example, the step 7 denotes entering the *if* statement, while the step 8 denotes the *out* statement. For passive agents, only statements inside procedures (i.e. inside curly brackets) are taken into consideration while counting steps. Note that statements related to *select* conditions (guards) are evaluated as step 2.

To simplify the formal description of transitions, for agent X we use the following notation convention:

- the target state of the described transition is denoted as $S' = (am'_X, pc'_X, ci'_X, pv'_X)$
- for a port p , p^* denotes a port associated with p in a communication diagram (note that $p = p^{**}$),
- we provide $nextpc_X$ function that determines a number of a next step (a next program counter for an agent) and $instr_X(i)$ function that determines a type of instruction associated with number of step, i .
- for a currently considered agent we neglect the postfix pointing an agent (i.e. we put a instead a_X).

The execution of the step i (called transition from state S to S') by the *running* agent changes a current state in the following way:

- if $instr(i)$ is **in p** then
 - if some other agent Y has put a message and is pending on port p^* then a message is taken form p and $pc' = nextpc$; note that the state of agent Y will also change form *waiting* to *running* and pc_Y points out an index of the next agent statement after *out* statement.

- if some other agent Y waits in a select guard containing **out p*** (i.e. $ci_Y = [q(\dots, \text{out } p^*: \text{npc}, \dots)]$) then the state S' is changed from *running* to *waiting* and $ci' = [\text{in } p]$; note that the state of agent Y will also change from *waiting* to *running*, $ci'_Y = []$ and $pc'_Y = \text{npc}$.
- if no agent waits on putting a message to the port p then the state S' is changed from *running* to *waiting* and $ci' = [\text{in } p]$.
- if $\text{instr}(i)$ is **out p** then we proceed analogously as in the previous case swapping *out* and *in* statements appropriately.
- if $\text{instr}(i)$ is a guard in **if g**, **elseif g** or **loop g** statements then the pc' is set dependently on the guard g value.
- if $\text{instr}(i)$ is a guard in **select g**
 - if the branch of guard g and was evaluated to true then we proceed as in the previous step; note that evaluating **in p** (**out p**) condition to true means that some agent Y has waited on the port p^* to send (receive) a message to (from) port p .
 - if no guard is evaluated to true but some branches, say b^1, \dots, b^k , can be evaluated to true if corresponding conditions (related to *in* or *out*) c_1^k, \dots, c_j^k change (in a result of activities of other agents) then the agent is shifted to the state S' , with $am = \text{waiting}$ and $ci = [g(c_1^k : \text{nextpc}^1 + \dots + c_j^k : \text{nextpc}^j)]$

5 Labelled Transition Graphs

Assume that $\overline{\mathbf{A}} = (D, B, \alpha^0)$ is an Alvis model. For the pair of states S, S' we say that S' is *directly reachable* from S iff there exists transition $t \in \mathcal{T}$ such that $S \xrightarrow{t} S'$. All states directly reachable from S is denoted as $\mathcal{R}(S)$. We say that S' is *reachable* from S iff there exists a sequence of states S^1, \dots, S^{k+1} and a sequence of transitions $t^1, \dots, t^k \in \mathcal{T}$ such that $S = S^1 \xrightarrow{t^1} S^2 \xrightarrow{t^2} \dots \xrightarrow{t_k} S^{k+1} = S'$. The set of all states that are reachable from the initial state S_0 is denoted by $\mathcal{R}^*(S_0)$.

States of an Alvis model and transitions between them are represented by a labeled transition system (LTS graph for short). An *LTS graph* is a directed graph $\text{LTS} = (V, E, L)$, such that $V = \mathcal{R}(S_0)$, $L = \mathcal{T}$, and $E = \{(S, t, S'): S \xrightarrow{t} S'\}$, where $S, S' \in \mathcal{R}(S_0)$ and $t \in L\}$. In other words, an LTS graph represents all states reachable from S_0 and transitions between them in the form of a directed graph.

Primarily we generate LTS graph for a single agent, starting from AlvisCL representation of its behavior. Let us consider the agent A presented in Listing 1.1 with LTS graph shown in Fig. 11. That agent contains most frequently used statements of AlvisCL.

The execution of the *loop* instruction is associated with an evaluation of condition and either execution of the first instruction of the loop block or execution of the first instruction after the loop. This situation is represented by states 0, 1 and 11. The execution of the *out p(x)* instruction is described by two states. The first one is associated with evaluation of agents environment and either this agent will be awaited (in the second state) or *out p* instruction can be executed

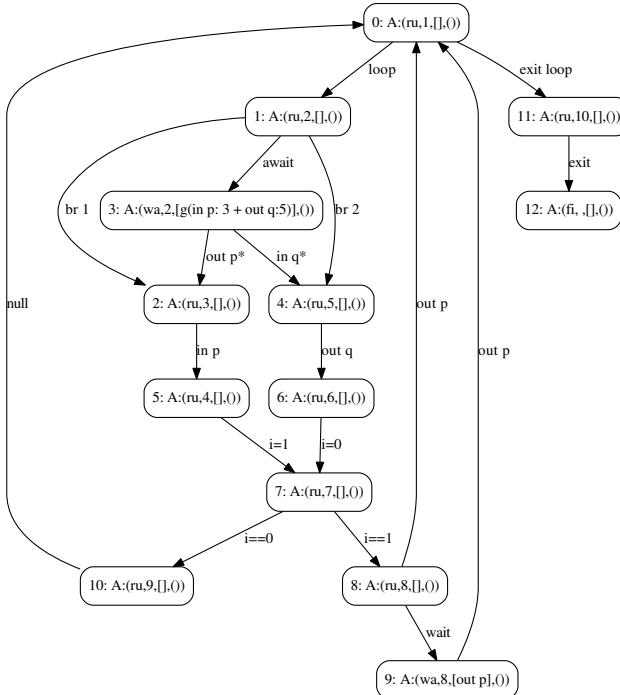


Fig. 1. LTS diagram for listing 1.1

(and we can move to the state associated with the next instruction). Note that the second case is possible if and only if some other agent has executed *in p(x)* instruction previously. This situation is represented by states 9 and 11.

In the case of *in* instruction we proceed similarly .

The execution of the *select* statement is associated with evaluation of the guard (state 1) and either execution one of the active branch (states 2 or 4) or waiting for an external event fulfilling the guard condition (described in the ci of the awaiting state (5)). Let us note that in the considered example *in* operations are represented by one state only (state 2 in the case *in p* and state 4 in the case *in q*), it is because the guards eliminates waiting.

The manual building of the LTS representing entire embedding system is very complex or even impossible, because we should consider thousands of states and transitions between them. For that reason it is necessary to develop the algorithm of automatic LTS generation.

6 LTS Generation

In this section we assume that we have a system consisting of N Alvis active agents X_1, X_2, \dots, X_N . Connections between them are represented by a communication diagram. Using the method described in the Section 4 we can generate

LTS for a single agent. Now we will show how to merge such individual LTS's into a composite one representing a whole system.

Let $S_{k,0}$ represents the initial state of the agent X_k then $\mathcal{R}^*(S_{k,0})$ represents states of LTS representing k -th agent. By C_i we denote cardinality of $\mathcal{R}^*(S_{k,0})$ set. By M we denote maximal number of states directly reachable from any state belonging to any single LTS. Now we can formulate two theorems.

Theorem 1. *The number of states of a composite LTS graph generated from LTS graphs A_1, \dots, A_N is not greater than $\prod_{i=1}^N C_i$.*

Proof. The proof is based on the observation that a number of possible states for a composite LTS is not greater than a number of cells in N -dimensional hypercube $\mathcal{HC} = \mathcal{R}^*(S_{1,0}) \times \mathcal{R}^*(S_{2,0}) \dots \times \mathcal{R}^*(S_{N,0})$ i.e. not greater than $\prod_{i=1}^N C_i$.

Theorem 2. *The complexity of a composite LTS graph generation from individual LTS A_1, \dots, A_N is limited by $\mathcal{O}(NM \prod_{i=1}^N C_i)$*

Proof. The general idea is using the hypercube \mathcal{HC} and putting edges inside it reflecting all possible subsequent transitions, starting from the initial state $S^0 = (S_{1,0}, S_{2,0}, \dots, S_{N,0})$. The unreachable states are removed.

The finding of all possible transitions is made in the Algorithm 1 and it is based on the observation that a transition in the hypercube \mathcal{HC} may be performed if some active agent, say A , is in the *running* state and it triggers a transition. Let consider it in a more detail.

The agent A being in *running* state may transite to following states:

1. *running*, when neither *in* nor *out* operation exists in a currently executed code line (Algorithm 1, line 1).
2. *running*, when either *in* or *out* operation is to be executed in a current code line and some active agent remains in a *waiting* state on suitable port, ready to contact (either write or read) A . In that case a waiting agent will also change its state to *running* (Algorithm 1, line 1).
3. *waiting*, when either *in* or *out* operation is in a current code line but no agent waits for A (Algorithm 1, line 1).

In Algorithm 2 we use the queue of composite states Q which initially is empty. To simplify pseudocodes the following notation was used in algorithms: $s \prec x, y$ denotes that for a given composite state s we replace individual states of given agents with their another states, x, y . States of other agents remain unchanged.

To evaluate the computational complexity of the Algorithm 2 we should remark that since each composite state can be enqueued at least once (as unvisited) the **while** loop (Algorithm 2, line 4) can be executed not more than $\prod_{i=1}^N C_i$ times. The loop **foreach** (Algorithm 2, line 9) is executed N times and in each case the size of S is increased by not more than M , hence $|S| \leq NM$. Thus body of the next loop **foreach** (Algorithm 2, lines from 12 to 15) can be executed not more than $NM \prod_{i=1}^N C_i$.

Algorithm 1. CheckTransition(x, s)

```

input :  $x$  – an individual agent's LTS state,  $s$  – a composite LTS state
output:  $S$  – set of all states accesible from  $s$ 
1 begin
2    $X \leftarrow$  the agent described by  $x$ ;
3   if no in/out in the current statement then
4     foreach running state  $x'$  directly reachable from  $x$  do
5        $s' \leftarrow s \prec x'$ ;
6        $S \leftarrow S \cup \{s'\}$ ;
7   else if current statement contains in/out and some agent  $Y$  waits for  $X$ 
then
8      $y \leftarrow$  current state of  $Y$ ;
9      $x' \leftarrow$  running state directly reachable from  $x$ ;
10     $y' \leftarrow$  running state directly reachable from  $y$ ;
11     $s' \leftarrow s \prec x', y'$ ;
12     $S \leftarrow \{s'\}$ ;
13   else if current statement contains in/out and no agent waits for  $X$  then
14      $x' \leftarrow$  waiting state directly reachable from  $x$ ;
15      $s' \leftarrow s \prec x'$ ;
16      $S \leftarrow \{s'\}$ ;
17   return  $S$ 
18 end

```

7 Example

To illustrate LTS graph generation we consider the model shown in Fig. 2 that represents two active agents that communicate one with another. The agent A is the sender and B is the receiver. The LTS graph for this model is shown in Fig. 3.

Initially we are in the state 0 defined as $X_1:(\text{running}, 1, [], []), X_2:(\text{running}, 1, [], [])$. If X_1 is running we move to the state 1 else we move to state 2. In both cases we add appropriate edges to the set EDGES and states 1, 2 to Q . In the state 1, the transition made with respect to the running agent X_1 shifts us to the state 5 defined as $X_1:(\text{waiting}, 2, \text{out}(p), [], []), X_2:(\text{running}, 1, [], [])$.

In this state only agent X_2 can run so we can move to state 7 defined as $X_1:(\text{waiting}, 2, \text{out}(p), [], []), X_2:(\text{running}, 2, [], [])$; the execution of **in q** shifts agent

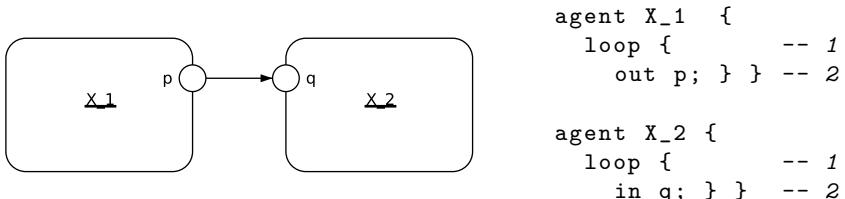


Fig. 2. Example 1

Algorithm 2. Merge($x_1^{(0)}, x_2^{(0)}, \dots, x_N^{(0)}$)

```

input :  $s^0 = (x_1^{(0)}, x_2^{(0)}, \dots, x_N^{(0)})$  – a sequence of individual initial states of
agents  $X_1, X_2, \dots, X_N$ 
output:  $G = (V, E)$  – a composite LTS graph for  $X_1, X_2, \dots, X_N$ 

1 begin
2    $Q \leftarrow s^0$ ;
3   Mark all composite states as unvisited;
4   while  $Q$  is nonempty do
5      $s \leftarrow Q.\text{dequeue}()$ ;
6     Mark  $s$  as visited;
7     Add  $s$  to  $V$  if not present;
8      $S \leftarrow \emptyset$ ;
9     foreach running state  $x$  in  $s$  do
10     $S \leftarrow S \cup \text{CheckTransition}(x, s)$ ;
11     foreach  $s' \in S$  do
12       Enqueue  $s'$  in  $Q$  if unvisited;
13       if  $s' \notin V$  then
14          $V \leftarrow V \cup \{s'\}$ ;
15        $E \leftarrow E \cup \{(s, s')\}$ ;
16    $G = (V, E)$ ;
17   return  $G$ ;
18 end

```

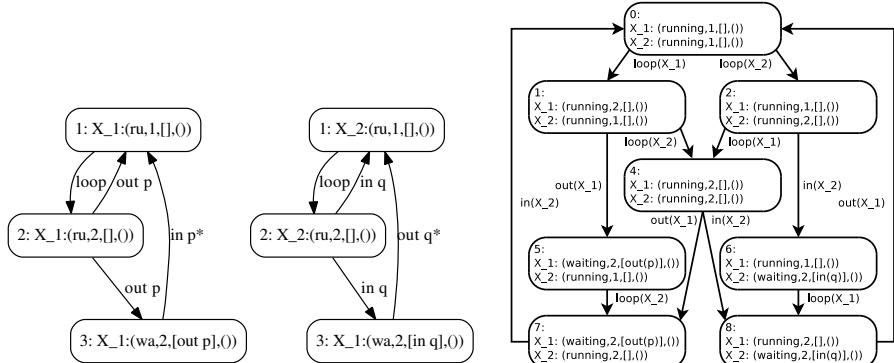


Fig. 3. a) LTS for produce and consumer b) Example 1 – LTS graph.

X_2 to state 1 or 3 in his individual LTS, in the context of the state 7 only first possibility may occur because X_1 is already waiting on the port **p**. In the context of the composite state 4 execution of the same operation, **in q**, leads X_2 to its "individual" state 3. In that case we move from composite state 4 to 8: $X_1:(\text{running},2,[],[]), X_2:(\text{waiting},2,\text{in}(q),[],[])$.

The rest of algorithm is the similar.

8 Summary

In the paper the algorithm of generation of the LTS for all agents present in a given system coded in the Alvis toolkit is presented. This generation is made in two phases: firstly we generate LTS graphs for single agents. In the second step we merge those graphs into a formal representation of a whole system. This gives the possibility of the formal verification of the defined system properties. The estimations of the space and computational complexities of this approach are also presented.

References

1. Szpyrka, M., Matyjasik, P., Mrowka, R.: Alvis – modelling language for concurrent systems. In: Bouvry, P., Gonzalez-Velez, H., Koodziej, J. (eds.) Intelligent Decision Systems in Large-Scale Distributed Environments. SCI, Springer, Heidelberg (2011)
2. Szpyrka, M., Matyjasik, P., Mrowka, R., Kotulski, L., Balicki, K.: Formal introduction to Alvis modelling language. International Journal of Applied Mathematics and Computer Science (to appear, 2011)
3. Garavel, H., Lang, F., Mateescu, R., Serwe, W.: CADP 2006: A toolbox for the construction and analysis of distributed processes. In: Damm, W., Hermanns, H. (eds.) CAV 2007. LNCS, vol. 4590, pp. 158–163. Springer, Heidelberg (2007)
4. Balicki, K., Szpyrka, M.: Formal definition of XCCS modelling language. Fundamenta Informaticae 93(1-3), 1–15 (2009)
5. Matyjasik, P.: Design and analysis of embedded systems with XCCS process algebra. PhD thesis, AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, Computer Science and Electronics, Kraków, Poland (2009)
6. Milner, R.: Communication and Concurrency. Prentice-Hall, Englewood Cliffs (1989)
7. Aceto, L., Ingófsdóttir, A., Larsen, K., Srba, J.: Reactive Systems: Modelling, Specification and Verification. Cambridge University Press, Cambridge (2007)
8. Barnes, J.: Programming in Ada 2005. Addison-Wesley, Reading (2006)
9. Burns, A., Wellings, A.: Concurrent and real-time programming in Ada 2005. Cambridge University Press, Cambridge (2007)

Snapshot Reachability Graphs for Alvis Models*

Marcin Szpyrka and Leszek Kotulski

AGH University of Science and Technology,
Department of Automatics,
Al. Mickiewicza 30, 30-059 Krakow, Poland
`{mszpyrka, kotulski}@agh.edu.pl`

Abstract. An embedded system usually consists of a set of sensors cooperating with one or more decisions centres. The design of such a system complicates in respect of both a complicated scheme of components interconnections and their parallel execution. In practice, the latter one excludes testing as a way to guarantee an expected level of a system quality. Thus, a formal verification of such systems is necessary. Alvis is a novel modelling language designed especially for embedded systems. However, it can be used for modelling any information system with concurrent activities. The key concept of Alvis is an *agent* that denotes any distinguished part of the system under consideration with defined identity persisting in time. The behaviour of agents is defined using Alvis Code Language (AlvisCL) that resembles high level programming languages. Interconnections among agents are defined using Communication Diagrams (AlvisCD) – a visual hierarchical modelling notation. For formal verification purposes, an LTS graph (Labelled Transition System) is generated for an Alvis model. The paper deals with the problem of encoding time relationships with LTS graphs if a model with agents that run concurrently is considered. As a solution, snapshot reachability graphs are proposed.

1 Introduction

The Phenomena, such as concurrency and non-determinism that are central to modelling embedded or distributed systems, turn out to be very hard to handle with standard techniques, such as peer reviewing or testing. Formal methods included into the design process may provide more effective verification techniques, and may reduce the verification time and system costs. Especially, the cost of improving of the software elements in an embedded system is unreasonable high. Thus, it is necessary to pay a special attention on a formal verification of the developed embedded systems.

Classical formal methods like Petri nets [1], [2], [3], [4], [5], process algebras [6], [7], [8], [9] or time automata [10], [3] provide techniques for a formal specification and modelling of concurrent systems but due to their specific mathematical syntax, they are very seldom used in real IT projects. In the presented approach, an embedded system is modelled with Alvis – a novel modelling language [11] designed by our team.

In the case of a single processor hardware environment, where concurrency is introduced by the time sharing technique, the standard LTS graph is sufficient for Alvis

* The paper is supported by the Alvis Project funded from 2009-2010 resources for science as a research project.

model verification. In such case, we can guarantee atomic execution of Alvis statements, thus only one of them is executed in a given state and we have no problem with designation of the next LTS node. The problem arises if multiprocessor environment is considered, because one of the executed operations can finish before the other ones. Thus, we cannot define a state in an LTS graph that denotes a situation with one operation (transition) completely executed and others only partially. In this paper we introduce *snapshot reachability graphs* that enable us to describe the behaviour of a model in a multiprocessor environment.

The scope of the paper is as follows. A short comparison of Alvis with other programming languages used for embedded systems development is given in Section 2. Section 3 presents time aspects of Alvis statements. Section 4 describes the idea of snapshot reachability graphs used for verification of time relationships in Alvis models. A short summary is given in the final section.

2 Alvis and Other Programming Languages for Embedded Systems

From the Alvis point of view, a system is seen as a set of agents that usually run concurrently, communicate one with another, compete for shared resources etc. Each agent has assigned a set of ports used for a communication with other agents or, if embedded systems are considered, with the corresponding system environment. Ports are used both to collect data (e.g. sensors reading) and to provide results of the agent activity (e.g. control signals for external devices). The behaviour of an agent is defined with AlvisCL statements. If necessary, a rule-based system can be encoded in Haskell and used to make decisions [11]. From the behaviour description points of view, agents are treated as independent individuals and defined components that can be used to compose a concurrent system. Communication diagrams are used to point out pairs of ports that make up communication channels used to exchange information between agents.

For the effective modelling, Alvis communication diagrams enable distributing parts of a diagram across multiple subdiagrams called *pages*. Pages are combined using the so-called *substitution mechanism*. An active agent on one level can be replaced by a page on the lower level, which usually gives a more precise and detailed description of the activity represented by the agent. Such a substituted agent is called *hierarchical* one. On the other hand, a part of a communication diagram can be treated as a module and represented by a single agent on the higher level. Thus, communication diagrams support both *top-down* and *bottom-up* approaches.

Alvis has its origins in the CCS process algebra [8], [9] and the XCCS language [12], [13]. The main result of the fact is the communication model used in Alvis that is similar to the one used in CCS and the rendes-vous mechanism used in Ada [14]. However, Alvis uses a simplified rendez-vous mechanism with equal agents without distinguishing servers and clients. A communication between two active agents can be initialised by any of them. The agent that initialises it, performs the *out* statement to provide some information and waits for the second agent to take it, or performs the *in* statement to express its readiness to collect some information and waits until the second agent provides it. In contrast to Ada, Alvis does not support asynchronous procedure calling,

a procedure uses always an active agent context. Finally, Alvis in contrast to Ada uses significantly less language statements and enables a formal verification.

A few constructs in Ada were an inspiration while developing Alvis language. For example, protected objects have been used to define passive agents and the Ada *select statement* has been used to define the Alvis *select statement*. An Alvis model composed of few agents that work concurrently is similar to an Ada distributed system. Active agents can be treated as processing nodes, while passive agents as storage ones.

Alvis has also many features in common with E-LOTOS – an extension of the LOTOS modelling language [15]. Alvis, like E-LOTOS, was intended to allow a formal modelling and verification of distributed real-time systems. For a given Alvis model an LTS graph can be generated that provides information about all reachable states of the considered system and transitions among them. Such graphs also provide a formal semantic for Alvis models and make a formal verification possible. For example, the CADP toolbox [16] and model checking techniques can be used to check whether a given model satisfies its requirements. Moreover, CADP offers a wide set of functionalities, ranging from step-by-step simulation to massively parallel model-checking.

In contrast to E-LOTOS, Alvis provides a graphical modelling language. Communication diagrams are the visual part of Alvis. They are used to represent the structure of the system under consideration. A communication diagram is a hierarchical graph that nodes may represent both kinds of agents (*active* or *passive*) and parts of the model from the lower level. They are the only way, in Alvis, to point out agents that communicate one with another. Moreover, the diagrams allow programmers to combine sets of agents into modules that are also represented as agents (called *hierarchical agents*).

Alvis has also many features in common with System Modelling Language (SysML) [17] – a general purpose modelling language for systems engineering applications. It contains concepts similar to SysML ports, property blocks, communication among the blocks and hierarchical models. Unlike SysML, Alvis combines structure diagrams (block diagrams) and behaviour (activity diagrams) into a single diagram. In addition, Alvis defines formal semantics for the various artifacts, which is not the case in SysML.

Due to the use of Ada origins, VHDL [18] and Alvis have a similar syntax for the communication and parallel processing. The concept of an agent in Alvis is also similar to a design entity in VHDL and both languages use ports for a communication among system components. It should be noted, however, that Alvis is closely linked with its graphical model layer. Graphical composition allows for easier identification of the system hierarchy and components. The main purpose of VHDL is the specification of digital electronic circuits and it focuses on systems hardware. However, Alvis integrates the hardware and software views of an embedded system.

In contrast to synchronous programming languages like Esterel [19], [20] or SCADE [21], Alvis does not use the broadcast communication mechanism. Only agents connected with communication channels can communicate one with another.

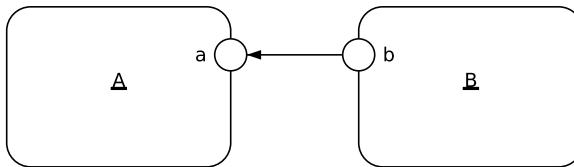
3 Time in Alvis

Alvis [11], [22] is a novel modelling language designed for embedded systems with origins in the CCS process algebra [8], [9], and the XCCS modelling language [12], [13].

In contrast to process algebras, Alvis uses a high level programming language based on the Haskell syntax [23], instead of algebraic equations, and provides a hierarchical graphical modelling for defining interconnections among agents.

AlvisCL provides carefully selected set of statements sufficient to describe a behaviour of individual agents. Each of them can have a duration assigned. For example, this is useful to evaluate how much time is necessary to provide a system answer for a given input event. Moreover, Alvis uses statements that use time explicitly:

- **delay t** – postpones an agent for a given number of time-units (default milliseconds);
- **alt(delay t) { . . . }** – defines a branch of the *select* statement that is open after the given number of time-units;
- **loop (every t) { . . . }** – repeats its contents (i.e. starts executing its contents) every specified number of time-units.



```

1  agent A {
2      loop {                                -- 1
3          select {                      -- 2
4              alt (ready [in(a)]) {
5                  in a;                   -- 3
6                  delay 1; }           -- 4
7              alt (delay 2) {
8                  null; }           -- 5
9      } } }
10 agent B {
11     loop (every 10) {                -- 1
12         out b; }                     -- 2
13 }

```

Fig. 1. Alvis model example

Unlike commonly used synchronous languages like Esterel, SyncCharts or SCADE, where emitted signals are accessible for any component, Alvis uses communication channels that join only two agents. This approach, taken from the CCS process algebra, is similar to the communication between tasks or a task and a protected object in the Ada programming language. Let us consider the Alvis model shown in Fig. II. The model contains two agents *A* and *B*. Each of them contains one port used for the communication with the other agent.

The example illustrates the three Alvis statements that use time explicitly. The *delay* statement (line 6) is used to postpone an agent for a given number of milliseconds. On the other hand, the *delay* guard (condition – line 7) used as a part of the *select* statement allows defining time-outs. The Alvis *select* statement is similar to the basic

select statement in the Ada programming language, but there is no distinction between a server and a client. The statement may contain a series of *alt* clauses called *branches*. Each branch may be guarded. These guards divide branches into *open* and *closed* ones. A branch is called *open*, if it does not have a guard attached or its guard evaluates to *True*. Otherwise, a branch is called *closed*. To avoid indeterminism, if more than one branch is open the first of them is chosen to be executed. If all branches are closed, the corresponding agent is postponed until at least one branch is open. If a branch contains *delay* as its guard then it is open after the given number of milliseconds. Thus, if all branches are closed, the corresponding agent waits the specified number of milliseconds and follows the branch. However, if at least one branch is open before the delay goes by, then the delay is cancelled. The last time statement is the *loop* statement with *every* guard (line 11). Such a loop repeats its contents every specified number of milliseconds.

4 Snapshot Reachability Graphs

Beside the graphical and code layer, an Alvis model contains the *system layer*. The third layer is predefined and depends on the model running environment i.e. the hardware and/or operating system. It is necessary for a model simulation and analysis. From the users point of view, the layer is the predefined one and it works in the read-only mode. Agents can retrieve some data from the layer, but they cannot directly change them. The system layer provides some functions that are useful for the implementation of scheduling algorithms or for retrieving information about other agents states. An example of such a system layer function is the *ready* function that takes as its argument a list of ports names of a given agent (with *in* or *out* keywords to point out the communication direction), and returns *True* only if at least one of these ports can be used for a communication immediately.

A user can choose one of a few versions of the layer and it affects the model semantic. System layers differ about the scheduling algorithm and system architecture mainly. There are two approaches to the scheduling problem considered. System layers with α symbol provide a predefined scheduling function that is called after each step automatically. On the other hand, system layers with β symbol do not provide such a function. A user must define a scheduling function himself.

The models considered in the paper use the α^0 system layer. This layer makes Alvis an universal formal modelling language similar to Petri nets, time automata or process algebras. The α^0 layer scheduler is based on the following assumptions.

- Each active agent has access to its own processor and performs its statements as soon as possible.
- The scheduler function is called after each statement automatically.
- In case of conflicts, agents priorities are taken under consideration. If two or more agents with the same highest priority compete for the same resources, the system works indeterministically.

A *conflict* is a state when two or more active agents try to call a procedure of the same passive agent or two or more active agents try to communicate with the same active agent.

A state of a model is represented as a sequence of agents states. To describe the current state of an agent we need a tuple with four pieces of information: *agent mode* (*am*), *program counter* (*pc*), *context information list* (*ci*) and *parameters values tuple* (*pv*). Due to the consider example let us focus on active agents only. A detailed description of agents states can be found in [1].

If α^0 system layer is considered, an active agent can be in one of the following modes: *finished* (*F*), *init* (*I*), *running* (*X*) and *waiting* (*W*). The *init* mode means that an agent has not yet started its activity, while the *finished* one means that it has already finished. The *waiting* mode means that an active agent is waiting either for a synchronous communication with another active agent or for a currently inaccessible procedure of a passive agent, and the *running* mode means that an agent is performing one of its steps.

In the classical LTS approach, behaviour of an Alvis model is considered at the level of detail of single steps. Each language statement is seen as one or more steps. Let us focus on the agent *A*. There are 5 steps in one agent cycle: (1) entering the loop, (2) entering the select statement and choosing a branch, (3) *in* statement, (4) *delay* statement and (5) *null* statement (see comments in Fig. 1). The program counter points out the currently executing step or the next step to be executed. For agents in the *init* or *finished* mode *pc* is equal to 0.

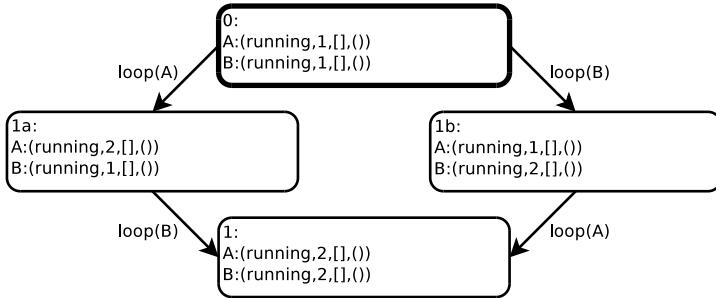
The *context information list* contains additional information about the current agent state. Suppose an agent *A* with port *p* is given. The context information list for the agent may contain the following items (non-exhaustive list):

- **in** (*p*) – *A* is waiting for a signal/value to be provided via the port *p* (by another agent or the environment);
- **out** (*p*) – *A* has provided a signal/value and is waiting for collecting it (by another agent or the environment);
- **timer** (*t*) – *A* is waiting for a timer signal that will be generated in *t* time-units (milliseconds by default);

In any state, *pv* contains the current values of the agent parameters (Agents in the considered example are parameterless).

Each of the possible steps is also described formally i.e. we define when a given step is *enable* and the result of its execution. Thus, starting from the initial state, we can generate the set of all states that are reachable from it and point out transitions (steps) leading from one state to another. Such a state space is usually represented using an LTS graph i.e. a directed graph with nodes representing reachable states and edges representing transitions among steps. Of course, different formal languages (e.g. Petri nets, time automata, process algebras) use different methods of describing nodes and edges in LTS graphs and uses different names for them (e.g. reachability graphs in Petri nets). Let us focus on LTS graphs generated for Alvis models. When time dependences are disregarded, we consider all possible interleaving of steps executed by agents. A small initial fragment of the LTS graph for the considered system is shown in Fig. 2. As you can see, each of the transitions *loop(A)* and *loop(B)* is considered separately. The graph does not express the fact that these steps can be executed in parallel.

Assume steps duration for all steps executed by the *A* and *B* agents as given in Table 1. When the α^0 system layer is consider and time values are important we

**Fig. 2.** Part of an LTS graph**Table 1.** Steps duration

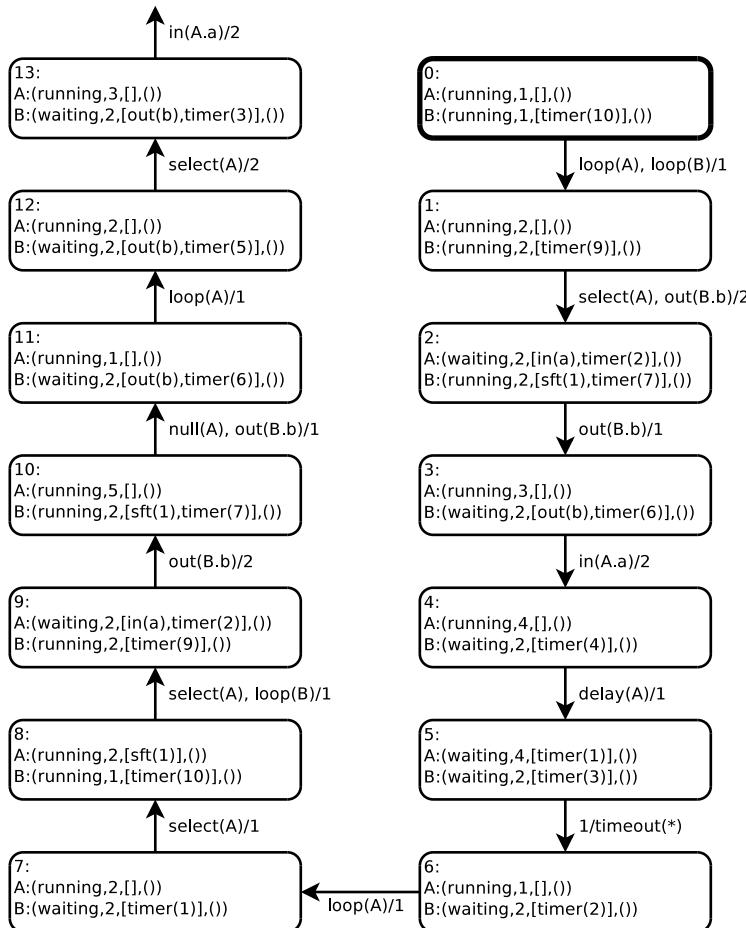
Agent A	Duration	Agent B	Duration
<i>loop</i>	1	<i>loop</i>	1
<i>select</i>	2	<i>out</i>	3
<i>in</i>	2		
<i>delay</i>	1		
<i>null</i>	1		

additionally assume the so-called *strong execution rule* i.e. an agent executes a step as soon as possible. It means that both agents start executing the *loop* steps in the same time and none of them can postpone the execution of it.

Let us focus on the initial state for the model under consideration (see Fig. 3 bold rounded box). Both agents are running their first steps (entering a loop). Moreover, *ci* for *B* points out that the next loop course starts in 10 ms. Thus, after 1 ms our system will be in the state 1 (see Fig. 3). Labels in the presented graph are of the form $t_1/transitions/t_2$, where t_1 stands for the time the system stays in the *old* state and t_2 stands for the duration of steps. If any of the time values is equal to 0, it is omitted together with the corresponding slash.

There are two transitions *select(A)* and *out(B.b)* enable in the state 1. (In case of *in* and *out* transitions the port is given, instead of an agent, to describe a transition precisely). Because the *select(A)* step takes 2 ms and *out(B.b)* takes 3 ms, we cannot present the result of these transitions execution as a state similar to the state 1. After 2 ms the step *out(B.b)* is still under execution, and after 3 ms the *A* agent is already executing another step. The solution for the problem is a *snapshot*. A *snapshot* is a state that presents the considered system with some steps under execution. We can take a snapshot every 1 millisecond, but we are interested only in these snapshots where at least one step has finished its execution. An LTS graph with snapshots will be called *snapshot reachability graph* or SR-graph for short. A part of the SR-graph for the considered model is shown in Fig. 3.

Let us go back to the problem of the successor for state 1. The next state in the SR-graph is as follows:

**Fig. 3.** Part of a snapshot reachability graph

```
A: (waiting, 2, [in(a), timer(2)], ())
B: (running, 2, [sft(1), timer(7)], ())
```

Agent A has already finished its $select(A)$ step and waits for either an input signal on the a port or for a time-out event because both branches are closed. Agent B is still running its $out(B.b)$ transition. The information $sft(1)$ (*step finish time*) means that it needs one millisecond to finish the current step.

Let us consider other elements of the SR-graph that relate to time. Let focus on the state 2. The A agent waits for either of the two events placed in its ci . One millisecond later (see state 3), the first branch is open, thus the agent executes steps the branch contains. A similar situation is in the state 9, but this time after 2 ms the second branch is chosen.

Let us consider the state 5. Both agents wait for a time-out event. The $(5, 6)$ edge label $1/timeout(*)$ means that this waiting lasts 1 ms and then as a result of time-out event generated by the environment ($*$ stands for a system environment) our system moves to the state 6.

5 Summary

A formal verification of time requirements for concurrent systems is a difficult task not always possible. Development of the Alvis modelling language has been carried out to guarantee such possibility. The snapshot reachability graphs described informally in this paper seem to meet the need of the formal analysis of time dependences.

It should be underlined that an SR-graph is strictly dependent on the steps duration. If we change the integers presented in Table 1 we will receive another SR-graph with possibly another paths. We can change the steps duration for the considered system in such a way that the second branch of the *select* statement will never be executed.

First of all, an SR-graph enable to check whether a given path (a sequence of steps) can be at all executed for a given steps duration. We can also determine the minimal and maximal times of passing between two given states, i.e. we can, for example, determine the maximal time of reaction of our system for an event. Moreover, SR-graphs enable us to verify all properties we can verify with standard LTS graphs, e.g. live-locks, deadlocks, process starvation etc. What is more important, the verification of these properties takes time dependences under consideration. For example, it is possible that we will find deadlocks in the standard LTS graph that are in fact not possible. In contrast to such an LTS graph, the corresponding SR-graph does not contain such deadlocks.

References

1. Murata, T.: Petri nets: Properties, analysis and applications. *Proceedings of the IEEE* 77(4), 541–580 (1989)
2. Jensen, K.: Coloured Petri Nets. In: *Basic Concepts, Analysis Methods and Practical Use*, vol. 1-3. Springer, Heidelberg (1992)
3. Bengtsson, J., Yi, W.: Timed automata: Semantics, algorithms and tools. In: Desel, J., Reisig, W., Rozenberg, G. (eds.) *Lectures on Concurrency and Petri Nets*. LNCS, vol. 3098, pp. 87–124. Springer, Heidelberg (2004)
4. Szpyrka, M.: Analysis of RTCP-nets with reachability graphs. *Fundamenta Informaticae* 74(2-3), 375–390 (2006)
5. Samolej, S., Rak, T.: Simulation and performance analysis of distributed internet systems using TCPNPs. *Informatica (Slovenia)* 33(4), 405–415 (2009)
6. Bergstra, J.A., Ponse, A., Smolka, S.A. (eds.): *Handbook of Process Algebra*. Elsevier Science, Upper Saddle River (2001)
7. Hoare, C.A.R.: *Communicating sequential processes*. Prentice-Hall, Inc., Upper Saddle River (1985)
8. Milner, R.: *Communication and Concurrency*. Prentice-Hall, Englewood Cliffs (1989)
9. Aceto, L., Ingólfssdóttir, A., Larsen, K., Srba, J.: *Reactive Systems: Modelling, Specification and Verification*. Cambridge University Press, Cambridge (2007)
10. Alur, R., Dill, D.: A theory of timed automata. *Theoretical Computer Science* 126(2), 183–235 (1994)

11. Szpyrka, M., Matyasiuk, P., Mrówka, R.: Alvis – modelling language for concurrent systems. In: Bouvry, P., González-Vélez, H., Kołodziej, J. (eds.) Intelligent Decision Systems in Large-Scale Distributed Environments. SCI, vol. 362, pp. 315–341. Springer, Heidelberg (2011)
12. Balicki, K., Szpyrka, M.: Formal definition of XCCS modelling language. *Fundamenta Informaticae* 93(1-3), 1–15 (2009)
13. Matyasiuk, P.: Design and analysis of embedded systems with XCCS process algebra. PhD thesis, AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, Computer Science and Electronics, Kraków, Poland (2009)
14. Barnes, J.: Programming in Ada 2005. Addison Wesley, Reading (2006)
15. ISO: Information processing systems, open systems interconnection LOTOS. Technical Report ISO 8807 (1989)
16. Garavel, H., Lang, F., Mateescu, R., Serwe, W.: CADP 2006: A toolbox for the construction and analysis of distributed processes. In: Damm, W., Hermanns, H. (eds.) CAV 2007. LNCS, vol. 4590, pp. 158–163. Springer, Heidelberg (2007)
17. Object Management Group: OMG Systems Modeling Language (OMG Sys ML) (2008)
18. Ashenden, P.: The Designer's Guide to VHDL, 3rd edn., vol. 3. Morgan Kaufmann, San Francisco (2008)
19. Berry, G.: The Esterel v5 Language Primer Version v5 91. Centre de Mathématiques Appliquées Ecole des Mines and INRIA (2000)
20. Palshikar, G.: An introduction to Esterel. *Embedded Systems Programming* 14(11) (2001)
21. Esterel Technologies SA: Welcome to SCADE 6.0. (2007)
22. Szpyrka, M.: Alvis On-line Manual. AGH University of Science and Technology (2011), <http://fm.ia.agh.edu.pl/alvis:manual>
23. O'Sullivan, B., Goerzen, J., Stewart, D.: Real World Haskell. O'Reilly Media, Sebastopol (2008)

Merging Belief Bases by Negotiation

Trong Hieu Tran, Quoc Bao Vo, and Ryszard Kowalczyk

Swinburne University of Technology,
John Street, Hawthorn, Victoria, Australia 3122
`{hhtran, bvo, rkowalczyk}@swin.edu.au`
<http://www.swinburne.edu.au>

Abstract. Belief merging has been an active research field with many important applications. Many approaches for belief merging have been proposed, but these approaches only take the belief bases as inputs without the adequate attention to the role of agents, who provide the belief bases, thus the results achieved are merely ideal and difficult to apply in the multi-agent systems. In this paper, we present a merging approach based on the negotiation techniques. A new model is proposed in which agents gradually build their common belief base from the beliefs that they provide in each round of negotiation. A set of postulates is also introduced to characterize the logical properties of the merging results.

Keywords: Belief merging, Belief Negotiation.

1 Introduction

Belief merging has been an important research area within computer science. The problem of belief merging is stated as following: *Given a set of sources (as propositional belief bases) which may be jointly inconsistent, how to obtain a common belief base from the sources?*

The solution to this problem is relevant to the area of database when multiple databases need to be merged, or information retrieval when multiple sources of information need to be aggregated, and also to multi-agent systems where agents with different beliefs about a domain need to reach a consensus for coordinating their activities. Many particular merging approaches have been proposed, for example, belief merging with arbitration operators by Revesz [14], belief merging with weighted belief bases by Lin [12], belief merging with integrity constraints by Konieczny [9], belief merging by possibilistic logic framework by Benferhat [3], and belief merging with stratified bases by Qi [13]. In general, these approaches are more advanced than the belief combination approaches [112] in the sense that they consider not only the union of all belief bases but also their sources. However, in these approaches, the agents, providing the belief sources, are not taken into account in the merging process. All sources are also assumed to provide explicit and complete beliefs they hold, and the merging process is handled by an independent and impartial mediator. This assumption is generally too strong with respect to most multi-agent systems.

In addition to the above approaches, belief merging by negotiation has also been considered [4,5,15,7]. This approach is from a natural and human-like idea

when resolving the conflicts of a committee, i.e. when a group of people have some conflicting opinions, to achieve the consensus, we let them discuss and negotiate with each other. This approach is introduced by Booth in [45] as a two-stage belief merging process based on the Levy Identity in belief revision [10]. In the works, authors propose a general framework for belief merging by negotiation, and it is continued by pointing out a family of merging operators by Konieczny in [15]. These works do not strong enough to force the sources to weaken their belief bases in a cooperative way, i.e. some agents can use some tricks in order to preserve their own beliefs in the merging result. Besides, because of keeping the minimal change property, inherited from belief revision, [45] violate fairness property, the important one for belief merging. Another work related to this approach is introduced by Zhang [15] by considering the bargaining game problem in the spirit of belief merging. This work is based on the idea of aligning all belief bases in the lowest priority layer and then iteratively removing the lowest layers of belief bases until their remaining segments become jointly consistent or a disagreement situation arises. This approach may lose some useful beliefs in case the beliefs do not cause any conflict, but they are on some low layers that need to remove. A similar situation also takes place when belief merging is carried out on the profile of stratified belief bases in [13]. Moreover, all these works require the agents to expose all their own beliefs so it is difficult to apply for the multi-agent systems.

According to negotiation point of view, belief merging is a process in which some agents will make some concessions in theirs own belief bases to reach the consensus. The agents are assumed to be truthful, rational and cooperative, i.e. agents give their own true beliefs, desire to preserve their beliefs as much as possible and accept all beliefs from others, provided that they do not conflict with their own beliefs. However, the assumption that agents are self-interested is very common in multi-agent systems, if so they will try to reach as much utility (preserving their own beliefs) as possible. Therefore, two questions arising are how agents make the least concession just enough to reach consensus and how to obtain the fair merging result for all agents. The answers for these questions are the main aims of this paper.

The remaining of this paper is organised as follows. In the next section, some formal preliminaries are provided. Section 3 makes an outline of some main approaches in the belief merging by negotiation, and some evaluations of those approaches are presented. Our new approach is introduced in Section 4, in which we present a model for belief merging and a set of postulates to characterize the merging operators as well as some logical properties are proposed and analysed. Finally, the conclusion is mentioned in Section 5.

2 Formal Preliminaries

We consider a propositional language \mathcal{L} over a finite set alphabet \mathcal{P} . \mathcal{W} denotes the set of possible worlds where each possible world is a function from \mathcal{P} to $\{\text{T}, \text{F}\}$ (T denotes the truth value true and F denotes the values false). The set of all subsets of \mathcal{W} is denoted by \mathcal{B} . A model of a formula ϕ is a possible world

ω which makes the formula true. We use $mod(\phi)$ to denote the set of models of formula ϕ i.e. $mod(\phi) = \{\omega \in \mathcal{W} | \omega \models \phi\}$. \succcurlyeq is a binary relation on a non-empty set X of \mathcal{L} . \succcurlyeq is a total pre-order if for all $\alpha, \beta, \gamma \in X$ we have i) $\alpha \succcurlyeq \alpha$, ii) if $\alpha \succcurlyeq \beta$, and $\beta \succcurlyeq \gamma$ then $\alpha \succcurlyeq \gamma$ and iii) $\alpha \succcurlyeq \beta$ or $\beta \succcurlyeq \alpha$.

A *belief base* K is a finite set of formulae which can be considered as the formula ϕ which is the conjunction of the formulae of K . Let K_1, \dots, K_n be n belief bases in which some of them may be similar, a *belief set* E of those n belief bases is a multi-set $E = \{K_1, \dots, K_n\}$. Suppose that $K = \{\phi_1, \dots, \phi_m\}$, we denote $\wedge K = \wedge_{i=1}^m \phi_i$ and $\wedge E = \wedge_{i=1}^n \wedge K_i$.

Two belief bases K and K' are logical equivalent, denoted $K \equiv K'$, if $\forall \phi \in K, K' \vdash \phi$ and vice versa. Belief set $E' = \{K'_1, \dots, K'_n\}$ is logical equivalent to belief set E , denoted $E \equiv E'$, if and only if there exists a permutation π such that $K_i \equiv K'_{\pi(i)}$ for all $i = 1, \dots, n$. The union of two belief sets E and E' is $E \sqcup E' = \{K_1, \dots, K_n, K'_1, \dots, K'_n\}$

We consider a set of agents $Sources = \{1, \dots, n\}$, each agent i has an *ordered belief base* (X_i, \succcurlyeq_i) in which $X_i \subseteq \mathcal{L}$, and $\succcurlyeq_i \subseteq X_i \times X_i$ is a total pre-order.

A *belief profile* is a multi-set of ordered belief bases. The set of all belief profiles from set of agents $Sources$ in language \mathcal{L} is denoted by $g^{Sources, \mathcal{L}}$.

Given a belief profile $G = ((X_i, \succcurlyeq_i))_{i \in Sources} \in g^{Sources, \mathcal{L}}$, a possible outcome is a tuple (O_1, O_2, \dots) such that $O_i \subseteq X_i$ for all $i \in Sources$. The set of all possible outcomes of belief profile G is denoted $\mathcal{O}(G)$.

For any two possible outcomes $O = (O_1, O_2, \dots)$ and $O' = (O'_1, O'_2, \dots)$ of a belief profile, we say that O *dominates* O' , denoted by $O \gtrsim O'$, if and only if $O'_i \subseteq O_i$ for all $i \in Sources$. O is *strongly dominates* O' , denoted by $O > O'$, if and only if $O \gtrsim O'$ and $O'_i \subset O_i$ for some $i \in Sources$. The Pareto set $\bar{\mathcal{P}}(G)$ is defined based on the concept of domination:

$$\bar{\mathcal{P}}(G) = \{O \in \mathcal{O}(G) : \nexists O' \in \mathcal{O}(G) \text{ s.t. } O' > O\}$$

A (pseudo-)distance function $dist : 2^{\mathcal{L}} \times 2^{\mathcal{L}} \rightarrow \mathbb{R}^*$ is a function that satisfies: i) $dist(X, Y) = 0$ if and only if $X \equiv Y$; and ii) $dist(X, Y) = dist(Y, X)$, where $X, Y \in 2^{\mathcal{L}}$ are two set of formulae. $dist(X, Y)$ presents the difference between two sets of formulae X and Y by a non-negative real number. The difference degree between a belief profile $G = ((X_i, \succcurlyeq_i))_{i \in Sources}$ and its possible outcome $O = (O_1, O_2, \dots)$ is defined by a function:

$$diff(G, O) = \max_{i \in Sources} (dist(X_i, O_i)) - \min_{i \in Sources} (dist(X_i, O_i))$$

For any $O, O' \in \mathcal{O}(G)$, we say that O' is not fairer than O with respect to belief profile G , denoted by $O \triangleright_G O'$ if and only if $diff(G, O') \geq diff(G, O)$.

The negotiation solution is defined as follows:

Definition 1. A *negotiation solution* f is a function that assigns to a belief profile in $g^{Sources, \mathcal{L}}$ a possible outcome. It means that for any $G = ((X_i, \succcurlyeq_i))_{i \in Sources} \in g^{Sources, \mathcal{L}}$, $f(G) = ((f_i(G))_{i \in Sources}) \in \mathcal{O}(G)$, where $f_i(G) \subseteq X_i$.

The minimal inconsistent subset of a set of formulae is widely used in belief revision and belief merging approaches. It is defined as follows:

Definition 2. Let X and Y be sets of formulae, X is a minimal inconsistent subset of Y iff: i) $X \subseteq Y$; ii) X is inconsistent and iii) $\forall \phi \in X, X \setminus \{\phi\}$ is consistent.

We denote $MIS(Y)$ as the set of minimal inconsistent subsets of Y . Formally, $MIS(Y) = \{X | X \text{ is minimal inconsistent subset of } Y\}$.

In literature, many logical properties have been proposed by the sets of postulates [14,9,11] to characterise a belief merging operator. We introduce the set of postulates proposed in [9], which is used to characterize Integrity Constraints (IC) merging operators. According to that work, the IC merging operator is stated as follows:

Definition 3. Let E, E_1, E_2 be belief sets, K_1, K_2 be consistent belief bases, and μ, μ_1, μ_2 be formulae from \mathcal{L} . Δ is an IC merging operator if and only if it satisfies the following postulates:

- (IC0) $\Delta_\mu(E) \vdash \mu$.
- (IC1) if μ is consistent then $\Delta_\mu(E)$ is also consistent.
- (IC2) if $\wedge E \wedge \mu$ is consistent then $\Delta_\mu(E) = \wedge E \wedge \mu$.
- (IC3) if $E_1 \equiv E_2$ and $\mu_1 \equiv \mu_2$ then $\Delta_{\mu_1}(E_1) \equiv \Delta_{\mu_2}(E_2)$.
- (IC4) if $K_1 \vdash \mu$ and $K_2 \vdash \mu$ then $\Delta_\mu(\{K_1, K_2\}) \wedge K_1$ is consistent if and only if $\Delta_\mu(\{K_1, K_2\}) \wedge K_2$ is consistent.
- (IC5) $\Delta_\mu(E_1) \wedge \Delta_\mu(E_2) \vdash \Delta_\mu(E_1 \sqcup E_2)$.
- (IC6) if $\Delta_\mu(E_1) \wedge \Delta_\mu(E_2)$ is consistent then $\Delta_\mu(E_1 \sqcup E_2) \vdash \Delta_\mu(E_1) \wedge \Delta_\mu(E_2)$.
- (IC7) $\Delta_{\mu_1}(E) \wedge \mu_2 \vdash \Delta_{\mu_1 \wedge \mu_2}(E)$.
- (IC8) if $\Delta_{\mu_1}(E) \wedge \mu_2$ is consistent then $\Delta_{\mu_1 \wedge \mu_2}(E) \vdash \Delta_{\mu_1}(E) \wedge \mu_2$.

These postulates are discussed in detail in many other works [9,13,6,8], hence in this work we do not discuss more about them but only refer to them to evaluate our work as well as to compare it with some others.

3 Related Works

In this section, we introduce about two most closest related works to our work. The first work is based on the notion of Social Contraction function proposed by Booth [4,5] and the second one is based on the notion of Simultaneous Concession Solutions for bargaining game problem by Zhang [15]. Both of works have been presented in both axiomatic and constructive approaches, but due to the lack of paper space, we only present their constructive approaches in the following subsections.

3.1 Social Contraction Function and Belief Negotiation

In [4,5], a two-stage approach to the belief merging problem is proposed. The first stage weakens the individual pieces of information such that they are jointly consistent. The second stage trivially adds all the weakened information together to achieve the merging result. The works concentrate on the first stage by building a model for it as follows:

In those works, each source $i \in Sources$ provides an item of information in the form of a set $S_i \in \mathcal{B}$. A belief vector is an element of \mathcal{B}^n . We use $\vec{S}, \vec{S}^1, \dots$ to denote belief vectors with $\vec{S} = (S_0, S_1, \dots, S_n)$, $\vec{S}^1 = (S_0^1, S_1^1, \dots, S_n^1), \dots$. A belief vector \vec{S} is consistent if $\bigcap_{i \in Sources} S_i \neq \emptyset$, otherwise it is inconsistent. $\vec{S} \subseteq \vec{S}^1$ means $S_i \subseteq S_i^1$ for all $i \in Sources$.

Let Ω denote the set of all finite sequences of belief vectors. Given $\sigma = (\vec{S}^0, \dots, \vec{S}^m) \in \Omega$ we will say σ is increasing if and only if $\vec{S}^i \subseteq \vec{S}^{i+1}$ for all $i = 0, 1, \dots, m-1$. We define the set of sequences $\Sigma \subseteq \Omega$ by

$$\Sigma = \{\sigma = (\vec{S}^0, \dots, \vec{S}^m) \mid \sigma \text{ is increasing and } \vec{S}^m \text{ is inconsistent}\}$$

Then a choice function is defined as :

Definition 4. *Choice function is a function: $g: \Sigma \rightarrow 2^{Sources}$ such that:*

$$(g0a) g(\sigma) \neq \emptyset$$

$$(g0b) i \in g(\sigma) \text{ implies } S_i^m \neq \mathcal{W} \text{ (where } \sigma = (\vec{S}^0, \dots, \vec{S}^m))$$

And a weakening function is defined as :

Definition 5. *Weakening function is a function $\nabla_\sigma: Sources \rightarrow \mathcal{B}$ such that:*

$$(\nabla 0a) S_i^m \subseteq \nabla_\sigma(i)$$

$$(\nabla 0b) \nabla_\sigma(i) = S_i^m \text{ implies } S_i^m = \mathcal{W}$$

Lastly, the solution to a belief vector with respect to a belief negotiation model is defined as follows:

Definition 6. *The solution to a belief vector \vec{S} for a belief negotiation model (relative to Sources) $\mathcal{N} = \langle g, \{\nabla_\sigma\}_{\sigma \in \Sigma} \rangle$ is given by the function $f^N: \mathcal{B}^n \rightarrow \Omega$, defined as: $f^N(\vec{S}) = \sigma = (\vec{S}^0, \dots, \vec{S}^k)$ where (i) $\vec{S}^0 = \vec{S}$, (ii) k is minimal such that \vec{S}^k is consistent, and (iii) for each $0 \leq j < k$ we have, for each $i \in Sources$,*

$$S_i^{j+1} = \begin{cases} \nabla_{\sigma_j}(i) & \text{if } i \in g(\sigma_j) \\ S_i^j & \text{otherwise} \end{cases}$$

Finally, we use Δ^N to denote the merging operator defined from a negotiation solution f^N as follows:

$$\Delta^N(\vec{S}) = \bigcap_{i=1}^n S_i^k \quad (1)$$

Refer to set of postulates for belief merging under integrity constraint above, we have:

Proposition 1. Δ^N satisfies for (IC0), (IC1), (IC2), (IC3), (IC7), (IC8); it does not satisfy properties (IC4), (IC5), and (IC6).

3.2 Belief Merging Approach for Bargaining Game

A belief merging technique used to solve the bargaining game problem is proposed by Zhang [15]. In this work, each demand is represented as a logical formula in propositional logic, the set of demands of an agent is presented in an ordered

belief base, and the outcome of bargaining game is the result of merging process. The solution for the merging problem is based on the notion of Simultaneous Concession Solutions. In this subsection, we first introduce the approach in constructive way, and then we have some evaluation and analysis about it.

We consider a ordered belief base (X_i, \succ_i) , $(X_i^1, X_i^2, \dots, X_i^{L_i})$ is a partition (or stratification [13]) of X_i by relation \succ_i if and only

- (1) $X_i^l \subseteq X_i$ and $X_i^l \neq \emptyset$ ($1 \leq l \leq L_i$)
- (2) $X_i = \bigcup_{l=1}^{L_i} X_i^l$
- (3) $X_i^l \cap X_i^k = \emptyset \forall l \neq k$
- (4) for any $\phi \in X_i^k, \psi \in X_i^l, \phi \succ_i \psi$ if and only if $k > l$

We can extend the partition into an infinite sequence $\{X_i^l\}_{l=1}^{\infty}$ by assuming that $X_i^l = \emptyset$ when $l > L_i$. $\{X_i^l\}_{l=1}^{+\infty}$ is called the *hierarchy of the demand set* (X_i, \succ_i) and L_i is called the *height of the hierarchy*.

We use $X_i^{>k} = \bigcup_{l>k} X_i^l$. In particular, $X_i^{>0} = X_i$.

The *Simultaneous Concession Solution* is defined as follows:

Definition 7 ([15]). A negotiation solution F on $g^{Sources, \mathcal{L}}$ is the Simultaneous Concession Solution if for any belief profile $G = (X_i, \succ_i)_{i \in Sources}$,

$$F(G) = \begin{cases} (X_1^{>\mu}, \dots, X_n^{>\mu}) & \text{if } \mu < L \\ (\emptyset, \dots, \emptyset) & \text{otherwise} \end{cases}$$

where $L = \min_{i \in Sources} L_i$ and $\mu = \min\{k : \bigcup_{i=1}^{Sources} X_i^{>k}\}$ is consistent.

Let $\Delta^G(G) = \bigcup_{i=1}^{Sources} f_i^G(G)$ be the merging operator with respect to a Simultaneous Concession Solution f^G . Refer to set of postulates for belief merging under integrity constraint above, we have:

Proposition 2. If f^G is a Simultaneous Concession Solution, then Δ^G satisfies for (IC1), (IC2), (IC5), (IC7), and (IC8); it does not satisfy properties (IC0), (IC3), (IC4), and (IC6).

As we see above, this work is based on the following principles:

- The result of merging process is consistent. (P1)
- The upper comprehensive segments of each ordered belief base must be preserved. (P2)
- The disagreement is not allowed. (P3)
- In each round of the negotiation process, all ordered belief bases have to make some concession. (P4)

These principles are rational in the bargaining game point of view. However, if we consider them in the belief merging point of view, they present some following disadvantages. Firstly, this approach prefers comprehensiveness to the speciality of information. For instance, in *Example 1* in [15], *jobs* is special information (it is only in an ordered belief base, not conflict with others) but still be removed

because of (P2) and (P4). Further, in [3] the authors have a subtle approach for merging stratified belief bases, but it still omits some useful pieces of information because they have priority lower than some inconsistent ones. Secondly, an important element in belief merging, the number of participants, is not properly considered (because of (P3) and (P4)). Lastly, disagreement (P3) is not rational for belief merging because in bargaining game, the outcome of negotiation is the goals in future, so if the disagreement arises, it means that agents do not achieve the common goals (and they need to change the demands and re-negotiate), the negotiation process should be terminated. It is different in a belief merging situation, in which belief bases reflect a same existing real world (but because of some reasons, some of them may be conflict with others), the aim of belief merging is finding out the truth or as near the truth as possible. If the disagreement arises i.e. there exists a belief base which is inconsistent and eliminated completely, it just means that the belief base is totally wrong about the existing real world, and the belief merging process should be continued.

4 Approach for Belief Merging by Negotiation

In this section, we introduce a new approach for belief merging by negotiation. The idea of this approach is based on following principles. Firstly, we gradually build a consensus, the set of common beliefs, instead of weakening the belief bases of participants. It is different from and more convenient than other existing approaches in two aspects. First, we do not need to create the counters of belief bases to keep original ones. Second and more important, when a belief profile has some equivalent belief bases (in both syntactic and semantic levels), instead of doing in each of the belief bases, we choose one of them as a representative and do with this representative. Lastly, we do not weaken the whole belief base; we only concentrate in the inconsistent parts of information. By using the notion of minimal inconsistent subset, we first isolate all beliefs joining in the conflict issues, then through the negotiation process, we will minimize this set and remove it to achieve the result.

In this work, the negotiation for belief merging with respect to a belief profile is described as follows: we first construct the set of minimal inconsistent subsets (of beliefs) from the belief bases, then we organise rounds of negotiation. In each round, some agents are chosen; the chosen agents will submit their most current preferred beliefs, which is not included in the set of common beliefs, to put into this set. If some submitting beliefs and the set of common beliefs arise jointly conflicts, or if we remove the equivalent formulae with them from the remaining set of the minimal inconsistent subsets, it makes any minimal inconsistent set become empty, then they are removed, otherwise they are added in the set of common beliefs, and all beliefs in any remaining set of minimal inconsistent subsets that is equivalent to them will be removed. The negotiation process is terminated when agents submitted all their beliefs. Each round of negotiation changes the negotiation state to other, in which negotiation state is presented by the belief profile, the constructing set of common beliefs, and the set of (non-empty) remaining set of minimal inconsistent subsets.

4.1 Negotiation Model for Belief Merging

In this subsection, we introduce the constructive approach for belief merging by negotiation. The idea of approach is iteratively constructing the set of common beliefs from belief profile with the references to the set of minimal inconsistent subsets. In each negotiation round, some beliefs are chosen from belief profile to put into the set of common beliefs and some beliefs are removed from some minimal inconsistent subsets. Thus, at each round, a negotiation state arises. Formally, the belief state is defined as follows:

Definition 8. A state in the negotiation process is represented as a tuple (G, X^*, F) , in which $G = \{(X_i, \succ_i)\}_{i \in \text{Sources}}$ is a belief profile, X^* is the constructing set of common beliefs, and F is the remaining set of minimal inconsistent subsets of belief bases.

The set of all negotiation states are denoted by $S^{\text{Sources}, \mathcal{L}}$.

In each negotiation round, when a group of agents are chosen, they will weaken their belief bases according to the weakening function, which is defined as follows:

Definition 9. $w : 2^{\text{Sources}} \times S^{\text{Sources}, \mathcal{L}} \rightarrow S^{\text{Sources}, \mathcal{L}}$ is a weakening function.

$w(A, S) = S'$ means that when the set of agent A are currently chosen, they will change the negotiation state from S to S' . $w(A, S)$ is *updatable* if $w(A, S) \neq S$. When $w(A, S)$ is triggered, each agent $i \in A$ will choose his most preferred belief satisfying for conditions (c0a) and (c0b) to put into the set of common beliefs simultaneously, and they will removes all equivalent pieces of belief in remaining sets of minimal inconsistent subsets. The choice function is defined as:

Definition 10. $c : S^{\text{Sources}, \mathcal{L}} \rightarrow 2^{\text{Sources}}$ is a choice function that satisfies $c(S) \neq \emptyset$ implies $w(c(S), S)$ is updatable.

$c(S)$ implies the set of chosen agents corresponding to the current negotiation state S .

A group of agents are chosen if (i) the distance from their beliefs to the common set of beliefs is maximal, and (ii) they can choose the most preferred piece of their belief such that:

- (c0a) there are not equivalent or conflict beliefs in the set of common beliefs, and
- (c0b) in the remaining set of minimal inconsistent subsets, there are not subset with one element such that this element is equivalent with some belief that they intent to put into the set of common beliefs.

$c(S) = \emptyset$ means that we can not choose any more (the beliefs of all agents have been considered), the negotiation process should stop.

The procedure for belief merging by negotiation is presented as follows:

- **Input:** Belief profile $G = \{(X_i, \succ_i)\}_{i \in \text{Sources}}$.
- **Output:** The set of common beliefs built from negotiation process.

Begin

$$G = \{(X_i, \succ_i)\}_{i \in \text{Sources}}; \quad X^* = \{\}; \quad F = MIS(\bigcup_{i \in \text{Sources}} X_i);$$

$$S_0 = (G, X^*, F); i = 0;$$

While $c(S_i) \neq \emptyset$ do

```

 $S_{i+1} = w(c(S_i), S_i);$ 
 $inc(i);$ 
 $(G_i, X_i^*, F_i) = S_i;$ 
 $return X_i^*;$ 
End;

```

We call the belief merging operator constructed by this model as Negotiation Belief Merging operator.

4.2 Postulates and Logical Properties

In this subsection, we propose a characterisation of negotiation solution which is used to build the merging operator by negotiation i.e. we introduce a minimal set of properties that an negotiation solution has to satisfy to obtain the rationality. We consider the set of postulates for negotiation function f^M with respect to belief profile $G = ((X_i, \succ_i))_{i \in Sources}$ as follows:

- (IR) $f_i^M(G) \subseteq X_i, \forall i \in Sources$. (*Individual Rationality*)
- (CO) $\bigcup_{i \in Sources} f_i^M(G)$ is consistent. (*Consistency*)
- (CP) $\forall \phi \in \bigcup_{i \in Sources} X_i \setminus (\bigcup MIS(\bigcup_{i \in Sources} X_i)), \bigcup_{i \in Sources} f_i^M(G) \vdash \phi$. (*Cooperativity*)
- (PO) $f^M(G) \in \bar{\mathcal{P}}(G)$. (*Pareto Optimality*)
- (FR) $f^M(G) \triangleright_G O$, for all $O \in \mathcal{O}(G)$. (*Fairness*)

These postulates present the desirable properties for the negotiation function, and their intuitive meaning can be understood as follows: Postulate (IR) requires that after the negotiation process, the accepted beliefs of each agent can not exceed its initial belief base. Postulate (CO) states that the set of common beliefs after the negotiation should be consistent. If some belief does not join in any conflict situation, postulate (CP) assures that it will be in the negotiation result. Postulate (PO) states that the result of negotiation should be the Pareto optimisation i.e. no agent can improve its beliefs in the set of common beliefs without making others worse off. Postulate (FR) assures that the negotiation result should be the fairest possible outcome of negotiation.

From the set of postulates and the negotiation model for belief merging in the previous subsection, we have the representation theorem which is stated as follows:

Theorem 1. $\Delta^M(G) = \bigcup_{i \in Sources} f_i^M(G)$ is a Negotiation Belief Merging operator if and only if f^M satisfies (IR), (CO), (CP), (PO), and (FR).

Refer to set of postulates for belief merging under integrity constraint above, we have:

Proposition 3. if $\Delta^M(G)$ is a Negotiation Belief Merging operator, it satisfies for (IC0), (IC1), (IC2), (IC7), (IC8) and, it does not satisfy properties (IC3), (IC4), (IC5), (IC6).

We also have the relationship between our belief merging operator and the operator in Zhang's approach [15] as follows:

Proposition 4. if $\Delta^M(G)$ is a Negotiation Belief Merging operator, $\Delta^M(G) \vdash \Delta^G(G)$.

5 Conclusion

In this paper, a new approach for belief merging based on negotiation technique is presented. The idea of the approach starts up from a very natural way when a group of agents want to achieve a common agreement from their jointly inconsistent belief bases. This approach is based on the isolation of the conflict issues and focus on solving them by negotiation. By this way, it avoids losing useful information, and it is strong enough to force agents to take part in the negotiation process in the cooperative way i.e. the agents can only make some concessions in the set of conflict beliefs. The belief merging operators are characterised by a set of intuitive postulates, and it ensures that the merging results are rational for the multi-agent systems.

In this work, we define the weakening function simply by eliminating some beliefs from the set of inconsistent beliefs. More subtle techniques for inconsistency handling in weakening function will be proposed in the future work.

References

1. Baral, C., Kraus, S., Minker, J.: Combining multiple knowledge bases. *IEEE Trans. on Knowl. and Data Eng.* 3, 208–220 (1991)
2. Baral, C., Kraus, S., Minker, J., Subrahmanian, V.S.: Combining knowledge bases consisting of first order theories. In: Raś, Z.W., Zemankova, M. (eds.) *ISMIS 1991. LNCS*, vol. 542, pp. 92–101. Springer, Heidelberg (1991)
3. Benferhat, S., Dubois, D., Kaci, S., Prade, H.: Possibilistic merging and distance-based fusion of propositional information. *Annals of Mathematics and Artificial Intelligence* 34, 217–252 (2002)
4. Booth, R.: A negotiation-style framework for non-prioritised revision. In: *Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge, TARK 2001*, pp. 137–150. Morgan Kaufmann Publishers Inc., San Francisco (2001)
5. Booth, R.: Social contraction and belief negotiation. *Inf. Fusion* 7, 19–34 (2006)
6. Everaere, P., Konieczny, S., Marquis, P.: Conflict-based merging operators. In: *KR*, pp. 348–357 (2008)
7. Konieczny, S.: Belief base merging as a game. *Journal of Applied Non-Classical Logics* 14(3), 275–294 (2004)
8. Konieczny, S., Lang, J., Marquis, P.: Da2 merging operators. *Artif. Intell.* 157, 49–79 (2004)
9. Konieczny, S., Pérez, R.P.: Merging information under constraints: a logical framework, vol. 12, pp. 773–808 (2002)
10. Levi, I.: Subjunctives, dispositions and chances. *Synthese* 34, 423–455 (1977)
11. Liberatore, P., Schaefer, M.: Arbitration (or how to merge knowledge bases). *IEEE Trans. on Knowl. and Data Eng.* 10, 76–90 (1998)
12. Lin, J.: Integration of weighted knowledge bases. *Artif. Intell.* 83, 363–378 (1996)
13. Qi, G., Liu, W., Bell, D.A.: Merging stratified knowledge bases under constraints. In: *Proceedings of the 21st National Conference on Artificial Intelligence*, vol. 1, pp. 281–286. AAAI Press, Menlo Park (2006)
14. Revesz, P.Z.: On the semantics of arbitration. *International Journal of Algebra and Computation* 7, 133–160 (1995)
15. Zhang, D.: A logic-based axiomatic model of bargaining. *Artif. Intell.* 174, 1307–1322 (2010)

Semantic Distance Measure between Ontology Concept's Attributes

Marcin Pietranik and Ngoc Thanh Nguyen

Institute of Informatics, Wroclaw University of Technology,
Wybrzeze Wyspianskiego 27, 50-370, Wroclaw, Poland
{marcin.pietranik,ngoc-thanh.nguyen}@pwr.wroc.pl

Abstract. In this paper we present our work on ontology alignment. Careful literature research has brought us to the point where we have noticed vaguenesses of previously developed approaches. The basic problem is lack of interest in elementary building blocks of ontological concepts, which are concepts' attributes. In our work we concentrate on defining these atomic elements and analyzing how their characteristics can influence the whole process of aligning ontologies. We claim that expanding attributes with explicit semantics increases reliability of finding mappings between ontologies - designating partial alignments between concepts built around mapping their structures treated as a combination of attributes and their semantics can improve the amount of information that can be transformed thanks with found mappings. We believe that our approach moves the focus from aligning simple labels of concepts to aligning information about the real world they express.

1 Introduction

Providing automated method of knowledge acquisition and management in modern computer systems that are frequently distributed and placed in online environment, is a complicated and difficult task. It covers many different aspects of handling and processing raw data, further transforming it to the form, where it can be treated as a semantic description of some fragment of reality. These topics include creating common vocabularies and thesauruses, extracting information from relational databases or user interactions, generating inference rules etc. The eventual purpose is supplying a portable representation of knowledge that could be utilized independently from language or implementation.

What binds these listed tasks is a way of storing preprocessed information in structures called ontologies, which formally are definitions of distinctive features of domain of discourse. In [7] they are informally defined as explicit specifications of conceptualizations. In contemporary approaches ontologies are reduced to labeled graphs, which definitions are based on W3C OWL standard of implementing them in a computer environment. Careful analysis of this issue has left us with cogitation about ontology structures and information they can express.

Storing knowledge with ontologies is only a partial solution of providing portable method of knowledge management. The other equally important issue is developing a way of translating information between ontologies. This task

is called in literature *ontology alignment* and it is a widely discussed problem. The practical aspect of it is, for example, integrating two catalogs of online stores that group their products in nested categories along with relationships between them (such as recommendations). The issue here is to conveniently migrate product categorization present in one store to the structure applied in the other. Formerly developed solutions concentrated on designating symmetrical results, which were based on pairwise correspondences between ontologies. In our opinion this is improper approach- consider given online store example. The fact that particular product categorization can be easily integrated into another one, does not imply correct integration in the other direction.

Developing our previous ideas described in [13] we managed to distinguish three levels of granulation that are present in ontologies. Figure 1 contains brief illustration of our approach.

ONTOLOGY STRUCTURE		
CONCEPTS		ATTRIBUTES
CONCEPTS SEMANTICS	CONCEPT'S STRUCTURE SEMANTICS	VALUES

Fig. 1. Ontological Stack - Levels of Granularity

On top of the stack we have placed *the structure level* that contains information about relationships occurring between concepts. It can refer to hierarchical connections, equivalency, contradiction or variety of other links that can exist between them. Right below *the attribute-concept level* is placed, which holds data about all of the concepts and attributes extracted from ontologies, along with descriptions of their structures. By *concepts' semantics* we call formal specifications of basic, atomic features of concepts. *Values* of attributes contain possible states they can acquire. What associates these two elements is the level of *concept-attribute assignment*, which assigns logic sentences to attributes in the context of particular concept, such that the same attribute can have different semantics in different concepts. For example, imagine the attribute *Address* that can be included both in the concept *Person* and in the concept *Webpage*. Obviously these two assignments give heterogenous meanings to the same attribute. What is more, such assignments can influence the possible valuations of attributes.

In our work we concentrate on defining concepts' attributes and analyzing how their characteristics can influence the whole process of aligning ontologies. We claim that this new approach to ontology alignment, expanding attributes with explicit semantics and incorporating these pieces of information during align process, increases reliability of designated mappings. Previous methods did not consider the way that attributes can change when utilized within different concepts (as in *Address*, *Person*, *Webpage* example) and therefore they omitted possible vaguenesses of designated results. Our method focuses on calculating distances not only between concepts' label and concepts' attribute sets, but also distances between the semantics of attributes within concepts (the way how attributes can

alter their features). Partial alignments between concepts built around mapping their structures treated as a combination of attributes and their semantics can improve the amount of information that can be transformed thanks to found mappings. Therefore, our approach moves the focus from aligning simple labels of concepts to aligning information about the real world they model.

The article is organized as follows. Section 2 contains formal definition on top of which we base our work. In Section 4 we describe our method of calculating similarities between concepts' attributes. Section 5 gives overview of experimental environment that was implemented. Section 3 contains a brief overview of the work that has been done and described in literature. The last section gives a short description of our ideas for the future works and concise summary.

2 Basic Notions

In our work we concentrate on defining the ontology with regard to its' basic and atomic building blocks, which are concepts' attributes. We have noticed that in recently developed approaches to ontologies ([4] and [11]) this aspect is frequently omitted and instead of it they deliver complex definitions that are in the same time narrowed only to implementation aspect.

Treating [2] as a starting point we propose defining ontology on higher level of granularity. Consequently, we define ontology as a triple:

$$O = (C, R, I) \quad (1)$$

where C is a set of concepts, R is a set of relationships between concepts ($R \subseteq C \times C$) and I is a set of instances. Next, we define concept c from set C :

$$c = (Id^c, A^c, V^c) \quad (2)$$

in which Id^c denotes a concept label (an informal name of a concept), A^c is a set of attributes belonging to the particular concept and V^c is a set of domains of attributes from A^c . The triple c is called *concept's structure*.

We will call Real World or Universe of Discourse the tuple (A, V) in which $A = \bigcup_{c \in C} A^c$ and $V = \bigcup_{c \in C} V^c$. This framework contains all of the possible attributes along with all of their possible valuations.

In this point we have noticed the necessity of defining semantics of attributes and concepts, which would describe their properties. Initial work has been done in our previous article [3]. Following it, we assume existence of finite set S containing atomic descriptions of attributes' semantics. An element from set S is a basic description given in natural language. For example:

- "The First Name of a Person" for attribute *First_Name*
- "The Day of Birth" for attribute *Birth_Day*
- "The Tax Identification Number" for attribute *TIN*

L_s is the formal language, incorporating symbols from S and basic logic operators \neg, \vee, \wedge . L_s is a sublanguage of the sentence calculus language. For example, for

$S = \{s_1, s_2, s_3, s_4, s_5\}$ we can give set $L_s = \{s_1 \wedge s_2, s_3 \wedge \neg s_4, s_5\}$. We assume that all of the formulas from L_s are represented in conjunctive normal form. Thus, every formula can be rewritten as $(s_1 \vee s_2 \vee \dots \vee s_k) \wedge (s_{k+1} \vee s_{k+2} \vee \dots \vee s_{k+n}) \wedge \dots \wedge (s_{k+m} \vee s_{k+m+1} \vee \dots \vee s_{k+m+p})$ and can be treated as a set of clauses.

Definition 1. By semantics of attributes within concepts we call a partial function:

$$S_A : A \times C \rightarrow L_s \quad (3)$$

We assign logic sentences to attributes in the context of particular concept, such that the same attribute can have different semantics in different concepts.

For example, imagine the attribute *Address* that can be utilized both in the concept *Person* and the concept *Webpage*. Obviously despite the same name, these two attribute-concept assignments have different meanings. Note that these meanings (despite differences between them) share something in common - both express that an attribute can identify concept in some space (for example, in the city for concept *Person* or in the Internet for concept *Webpage*).

Definition 2. By concepts' semantics we define a function:

$$S_C : C \rightarrow L_s \quad (4)$$

This assignment let us give concepts particular meanings (analogically to attributes' semantics). Such approach allows us to distinguish concepts that share the same set of attributes, which share the same semantics. For example, the concept *Person* and *Building* can both utilize attributes *Name* and *Address*. Analyzing them only on the attribute level would lead to their indistinguishability.

Explicit assignments of semantics to both attributes and concepts allow for convenient identification of correspondences between concepts' attributes and concepts. Formally, we can provide unequivocal criteria for such connections.

Definition 3. Two attributes $a, b \in A$ are equivalent referring to their semantics (semantical equivalence) if the formula $S_A(a, c_i) \Leftrightarrow S_A(b, c_j)$ is a tautology for any two $c_i, c_j \in C(c_i \neq c_j)$.

Definition 4. The attribute $a \in A$ in concept $c_i \in C$ is more general than attribute $b \in A$ in concept $c_j \in C$ referring to their semantics (semantical generality) if the formula $S_A(a, c_i) \Rightarrow S_A(b, c_j)$ is a tautology for any two $c_i, c_j \in C(c_i \neq c_j)$.

Note that in general attributes a and b may be the same attribute in the set A . What differs them is the concept that they've been assigned to.

Definition 5. Two attributes $a, b \in A$ are in contradiction referring to their semantics (semantical contradiction) if the formula $\neg(S_A(a, c_i) \wedge S_A(b, c_j))$ is a tautology for any two $c_i, c_j \in C(c_i \neq c_j)$.

Definition 6. Two concepts $c_i, c_j \in C$ are equivalent referring to their semantics (semantical equivalence) if the formula $S_C(c_i) \Leftrightarrow S_C(c_j)$ is a tautology.

Definition 7. The concept $c_i \in C$ is more general than the concept $c_j \in C$ referring to their semantics (semantical generality) if the formula $S_C(c_i) \Rightarrow S_C(c_j)$ is a tautology.

Definition 8. Two concepts $c_i, c_j \in C$ are in contradiction referring to their semantics (semantical contradiction) if the formula $\neg(S_C(c_i) \wedge S_C(c_j))$ is a tautology.

Note that these three definitions above identify three types of relations from the set R . Also note that in R exist other relations (given explicitly by ontology designer) that cannot be described incorporating formal criteria.

The obvious doubt is developing new notation for representing ontologies and not utilizing available OWL standard (<http://www.w3.org/TR/owl-features/>). The main reason is lack of possibility of grounding the basic elements of expressing information within ontologies. Considering the *Webpage-Person-Address* example, the only way of avoiding the name ambiguity of the attribute *Address* is labeling it differently within different concepts. The disadvantage in this case is losing the opportunity to represent expressivity of natural language and therefore - losing the semantical content of ontologies, reducing them only to the syntactic level.

The second reason is vagueness when applying any kind of extension to the OWL notation (for example, developing markup that expresses attributes' semantics) - after incorporating such changes further processing of prepared OWL files will be difficult or even impossible due to the lack of support for made changes within available tools.

3 Related Works

Ontology alignment is widely discussed in literature. [4] describes the state of the art of this topic, including detailed investigation of both contemporary and former works. Fundamental approach (that is common to all solutions depicted in [4]) presents ontology alignment as a task of finding the set *Align* of tuples with following structure: (id, e, e', r, n) , where id is a unique identifier of a peculiar tuple from *Align*, e and e' are entities belonging to two different ontologies that are being mapped, r is a relationship connecting these entities (*equivalence*, *disjointness* and *generalization*). n is a confidence degree - a normalized similarity value ($n \in [0, 1]$) between two entities. Basically, it can be described as generating set of tuples consisting of identifiers of entities from different, heterogeneous ontologies, such that calculated similarities (with explicitly adopted method) between them cross given threshold value.

Another comprehensive approach to this topic can be found in [2]. This position includes elaborations on many different issues that appear during ontology alignment. Different chapters contain considerations about efficiency of ontology matching algorithms, evaluations of obtained results, merging different mappings or alignments evolution. Nevertheless, only few of them covers the grounding level of the subject, by analyzing the possible influence, that utilizing attributes may

have on the whole process, and none of them considers extending them with explicitly given semantics. Therefore, analyzed solutions lack the consistent methodology of treating ontologies. The second issue that we have noted the fact that definitions are all based on OWL, which is an implementation standard, that should be treated as the evaluation tool, not the foundation of provided formalisms. What is more-ontologies as the way of expressing knowledge should not be reduced to flat files, that are further processed, but kept intact.

In our work we need to include reliable method of calculating distances between two logic statements. After careful analysis of solutions to this problem, which were found in literature [1], [3], [9] we were able to identify basic approach that is mutual to all of them. This common methodology treats logic statements (assumed that they are expressed in conjunctive or disjunctive normal form) as sets of atomic symbols taken from shared alphabet, and then utilizes one of the many different functions calculating distances between finite sets, such as Jaccard's or Hamming's distance. In further parts of this article we will adopt following markup from [6]: for two statements $d1$ and $d2$ n is the number of propositional letters occurring only in $d1$, m is the number of propositional letters occurring only in $d2$, l the number of propositional letters occurring both in $d1$ and $d2$.

An interesting method of calculating distances between logic statements has been introduced in [10]. This approach is based on designating distances not only between raw sets of symbols, but also incorporating interpretations and models of particular formulas. Authors define models of a formula as a set of partial vectors containing truth values of elementary conjunctions combined with atoms from base alphabet that do not appear in processed conjunction. The standard Dalal's distance ([3]) is then applied. The more detailed description of this approach can be found in part 4 of this article.

In [6] a method of calculating distance (and identifying correspondence) between Horn's clauses has been introduced. Authors formulate general criteria that are developed to support such comparisons. The basic idea is built around observation that formerly created approaches (such as aforementioned Hamming's distance) returns maximal value of 1 when the two expressions share no feature in common (are entirely different). Therefore, they cannot identify the degree of dissimilarity between statements. Authors have developed a heuristic function that includes these considerations. According to intuition the distance value decreases when the number of common elements in two clauses raises. Thus, the initial equation does not return the minimal value of 0, when input statements are completely overlapping ($d1=d2$), but such case can be easily checked right before evaluating the function. This method is described wider in section 4.

4 Semantic Distance between Attributes and Concepts

Assigning explicit semantics to attributes and concepts give us possibility to designate the degree of similarity between any two attributes or concepts. Therefore, allowing to calculate the strength of such relationship and further - how tightly two attributes or concepts are connected.

Definition 9. By semantical distance we call a following function:

$$d_s : L_s \times L_s \rightarrow [0, 1] \quad (5)$$

This function must satisfy following conditions:

1. $d_s(d, d) = 0 \forall d \in L_s$
2. $d_s(d1, d2) = 0 \forall d1, d2 \in L_s$ if the formula $d1 \Leftrightarrow d2$ is a tautology
3. $d_s(d1, d2) = 1 \forall d1, d2 \in L_s$ if the formula $\neg d1 \wedge d2$ is a tautology
4. $d_s(d1, d2) = d_s(d2, d1) \forall d1, d2 \in L_s$

Henceforth, we can develop this function into two variations that are able to calculate distances between semantics of concepts and attributes:

$$d_C : C \times C \rightarrow [0, 1] \quad (6)$$

$$d_A : (A \times C) \times (A \times C) \rightarrow [0, 1] \quad (7)$$

As was previously said in section 3 we can utilize one of two in our opinion most reliable methods of designating distance between individual clauses. Assuming the same notation as in 3 the first method is taken from 6:

$$d_s(d1, d2) = \begin{cases} 0 & \text{iff } d1 = d2 \\ 1 - \frac{1}{2}(\frac{l+1}{l+n+2} + \frac{l+1}{l+m+2}) & \text{iff } d1 \neq d2 \end{cases} \quad (8)$$

The second method is incorporating the approach described in 10.

$$d_s(d1, d2) = \frac{\sum_{m \in Mod(d1)} \min_{m' \in Mod(d2)} Dist(m, m')}{|Mod(d1)|} \quad (9)$$

Where function *Mod* is defined as follows:

$$Mod(m) = (\overline{m}^+ \times \{l \mid l \in S \setminus m^+\}) \cup \overline{m}^+ \quad (10)$$

Where *Dist* is a straightforward Dalal's distance (3). Function *Mod* where \overline{m} is a set of all symbols from m and \overline{m}^+ is a set of all positive symbols from m . For details please refer to 10.

Having these tools we were able to develop the algorithm that calculates distance value between two attribute-concept semantic assignments. Assuming that both input semantics are given in conjunctive normal forms we can treat them as two sets of clauses $s1$ and $s2$ with fixed sizes respectively o and p . The function *dist* is calculated according to chosen method of calculating distance between individual clauses, that is, it may adopt the form of equation 8 or 9.

Algorithm 1 - Calculating distance between attributes' semantics

BEGIN

0. generate empty result table T with dimensions $o \times p$
 - avg_c = 0
 - avg_l = 0

```

1. for each clause i in s1
   for each clause j in s2
      T[i][j] = dist(s1[i],s2[j])
2. for each column i in T
   avg_c += minimal value from lines in i
3. avg_c = avg_c / o
4. for each line j in T
   avg_l += minimal value from columns in j
5. avg_l = avg_l / p
6. result = 0.5 * (avg_c + avg_l)
7. return result
END

```

Our algorithm has quadratic calculation complexity and it meets all of the criteria described at the beginning of the current section. The basic idea behind this algorithm is to provide flexible method of calculating distances between sets of clauses that appear to be the most convenient way of expressing semantics of attributes within concepts. The algorithm crawls through only one of the input sets and designates partial distances pairwise to elements of the second input. This approach ensures that both semantics are completely covered and no information loss occurs. Then it calculates average value for both minimal distances from the first set to the second and from the second to the first. This asserts that the final distance value includes not just shortest paths from one of the expressions, but incorporates both of the inputs. As aforementioned, such solution asserts it's own symmetry. What is remarkable, it is very easy to swap *dist* function to any other method and adjust it to fit different usages and execution conditions.

5 Implementation of Experimental Environment

In parallel with working on formal approach to ontology alignment described in section 2 we have developed the experimental environment that incorporates all of the definitions that have been elaborated. Figure 2 contains brief illustration of created system. It has layered architecture built around web available user interface. Basic idea was to provide convenient system that would be able to store, process and manage ontologies, represented in RDF (<http://www.w3.org/RDF/>) or OWL (<http://www.w3.org/TR/owl-features/>), temporarily restricted only to its former version 1.1.

The main functionality is placed in Core Engine - this module allow to process ontologies, extract concepts and their attributes from parsed OWL files and calculate different similarity measures or distances. These features are implemented using RDFLib, which is a Python library that provides convenient tools for parsing RDF files. To expand it's capabilities to be able to work with OWL files, we have decided to incorporate FuXiOWL package (<http://code.google.com/>

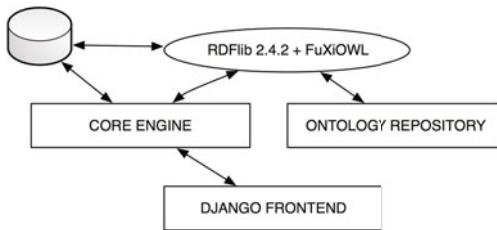


Fig. 2. Implementation Environment Architecture

p/fuxi/), which is a powerful plugin to RDFLib. Elements obtained from ontologies are stored in relational database - this solution yields effortless visualizing held structures. Ontologies itself are kept intact on the hard drive of the server.

Due to the fact that this system is still under intense development process, we have decided to incorporate few simplifying assumptions. As for now, attributes' semantics are reduced only to conjunctions of atoms. Whole dictionary of literals is also stored in a flat database table and does not provide any reasoning mechanisms. Similarity core engine is build around few basic functions as Levenshtein Distance, Dalal Distance and a function inspired by clause similarity framework described in [5] and overviewed in section 4. Because of limited space available for this article we will present experimental results of our approach and implemented system in other publication.

6 Future Works and Summary

In this paper we have developed further our work from [13]. We have expanded the ontology definitions with detailed description of attributes and how they change their characteristics when utilized in different concepts.

Due to the limited space available for this article we will present experimental results of our approach and implemented system in other publication. The main idea for conducting tests is to create the enviroment that handles uploading ontologies and allowing to assign semantics to attributes extracted from them. This part will be done manually. Then we will gather results from developed algorithm and comparing them with benchmarks available from OAEI (<http://oaei.ontologymatching.org/>). In the future we want to concentrate on extending our comparing attributes into a robust solution of designating distance between two concepts. This task will include incorporating not only distances between attributes, but also relationships that can occur between them. In parallel we will continue our work on creating standardized way of managing and processing ontologies within implemented system.

Acknowledgment. This research was partially supported by Polish Ministry of Science and Higher Education under grant no. 4449/B/T02/2010/39 (2010-2013).

References

1. Bisson, G.: Learning in FOL with a similarity measure. In: Proceeding AAAI 1992 Proceedings of the 10th National Conference on Artificial Intelligence, pp. 82–87. AAAI Press, Menlo Park (1992)
2. Bellahsene, Z., Bonifati, A., Rahm, E. (eds.): Schema Matching and Mapping, 1st edn. Springer, Heidelberg (2011)
3. Dalal, M.: Investigations Into a Theory of Knowledge Base Revision: Preliminary Report. In: Proceedings of the 7th National Conference on Artificial Intelligence, pp. 475–479 (1988)
4. Euzenat, J., Shvaiko, P.: Ontology Matching, 1st edn. Springer, Heidelberg (2007)
5. Ferilli, S., Basile, T.M.A., Biba, M., Di Mauro, N., Esposito, F.: A general similarity framework for horn clause logic. *Fundamenta Informaticae* 90(1), 43–66 (2009)
6. Ferilli, S., Biba, M., Basile, T., Di Mauro, N., Esposito, F.: k-Nearest Neighbor Classification on First-Order Logic Descriptions. In: Proceedings of IEEE International Conference Data Mining Workshops, ICDMW 2008, pp. 202–210 (2008)
7. Gruber, T.R.: A translation approach to portable ontology specifications. *Knowledge Acquisition* 5(2), 199–220 (1993)
8. Jung, J.J.: Ontology Mapping Composition for Query Transformation on Distributed Environments. *Expert Systems with Applications* 37(12), 8401–8405 (2010)
9. Lafage, C., Lang, J.: Propositional Distances and Preference Representation. In: Benferhat, S., Besnard, P. (eds.) ECSQARU 2001. LNCS (LNAI), vol. 2143, pp. 48–59. Springer, Heidelberg (2001)
10. Losada, D.E., Barreiro, A.: Efficient algorithms for ranking documents represented as dnf formulas. In: Proceedings SIGIR 2000 Workshop on Mathematical and Formal Methods in Information Retrieval, 1624, Athens, Greece (2000)
11. Maedche, A., Staab, S.: Ontology learning for the semantic web. *IEEE Intelligent Systems and Their Applications* 16(2), 72–79 (2001)
12. Nguyen, N.T.: Advanced Methods for Inconsistent Knowledge Management. Advanced Information and Knowledge Processing. Springer, Heidelberg (2008)
13. Pietranik, M., Nguyen, N.T.: Attribute Mapping as a Foundation of Ontology Alignment. In: Nguyen, N.T., Kim, C.-G., Janiak, A. (eds.) ACIIDS 2011, Part I. LNCS (LNAI), vol. 6591, pp. 455–465. Springer, Heidelberg (2011)
14. Staab, S., Studer, R.: Handbook on Ontologies, 2nd edn., vol. XIX, 811 p. 121 illus. Springer, Heidelberg (2009), Hardcover, ISBN: 978-3-540-70999-2

Casual Schedule Management and Shared System Using an Avatar

Takashi Yoshino and Takayuki Yamano

Wakayama University,
930 Sakaedani. Wakayama, 640-8510, Japan
yoshino@sys.wakayama-u.ac.jp
<http://www.yoslab.net/>

Abstract. Many organizations do not use schedulers, that is schedule management and shared systems. Often, this is because users find it difficult to remain motivated to use such systems. Therefore, we have developed a casual schedule management and shared system that makes use of an avatar. The dress-up items and accessories available for each avatar increase, depending on the number of items in a schedule. This system can be used on mixi—a Japanese social networking site. From the results of our experiments, we found that our system was able to maintain a user’s motivation. This is particularly true of users who are interested in acquiring dress-up items and accessories, as we found that they input their schedule more than the users who are less interested in those aspects of the system.

Keywords: schedule sharing, social network service, avatar, motivation.

1 Introduction

Traditionally, hand-written tools, such as a pocketbook and a calendar, are used more often for schedule management than digital tools, such as a cellular phone and a PC. However, with the spread of smart phones, such as the iPhone and Android-based phone, the number of users with digital schedule management tools has increased every year. The advantages of digital schedule management systems are the ease of sharing information, the ability to edit a schedule, and the variety of available browsers. The major disadvantage of digital tools is that the motivation to continue using them often decreases, partially owing to the time it takes digital tools to start up and run. Other reasons are that a scheduler made of paper is convenient and the shared advantages are not understood at once. According to the results of a survey in an Internet white paper in 2009 in Japan, only 10% of the respondents were using a schedule management and shared system. The ratio of the answer to “I am not using it though I know the tool” to “I do not do so now, but I have used the tool before” is higher than other Web-based systems. Schedule management tools tend not to be used as much as other Web systems.

As an alternative to conventional schedule management systems, services that use an avatar have now become popular on the Internet. Among these, Second

Life [1] is the most well known. An avatar is an animated representation of a computer user, and it can be a three-dimensional model, such as those used in computer games, or a two-dimensional picture, such as those found on Internet forums and other communities. As an example, “Nicotto town” is a Japanese community site in which a user can enjoy a virtual life by means of an avatar [2]. Other popular social networking sites in Japan include GREE and ameba pygmy. We believe that the use of avatar can provide the potential for taking as the work of daily life, especially to remain motivated.

On the basis of this background, we have developed a schedule management system called Character schedule. The purpose of this system is to perform schedule management and share that information more easily than traditional systems. In the Character schedule, the user obtains items to dress up an avatar according to the number of inputs to the schedule. Moreover, the action of the avatar changes in proportion to the number of inputs to the schedules, as well.

2 Related Works

There is some existing research on schedule management and sharing systems, as well as on using entertainment for motivation. However, there is little research that combines both of these aspects. In this section, we describe a research that has been conducted on maintaining motivation.

Kuramoto et al. researched how to maintain motivation by using entertainment. One of their systems is the Weekend Battle system [3]. The Weekend Battle system motivates employees to perform deskwork. The character in the system grows up according to the amount of effort a user puts in on a weekday. The purpose of the system is to improve motivation by means of a character fight on the weekend. In the first stage of the experiment, users did maintain and even improved their level of motivation. However, the level of motivation of users who had a weak character in the system gradually dropped. Then, Kuramoto et al. developed a new system called KAIYU-KAN [4]. In KAIYU-KAN, each user grows a tropical fish that reflects the amount of effort a user puts in during a week. Users compare fish based on “favor” rather than a “fight.” In a long-term experiment, they showed the effectiveness of the system in achieving the target of intuitive feedback and obtaining food for the fish.

Kuramoto et al. showed that it was effective to give users numerous options on how to express their fish. According to Chao, a beginner can improve his/her intuitive understanding of specialized work by means of the entertainment aspect of the system [5]. Bernhaupt et al. proposed a system that aims to maintain and improve social relationships [6]. Their system presents a worker’s expression in the office as a plant [6].

Research into using entertainment in a normal system has just begun. At this stage, a lot of the research and findings are compiled by trial and error. This study is one such and attempts to maintain and improve motivation by using an avatar in a schedule management system.

3 What Is a User’s Motivation in Schedule Management?

In this section, we describe our target user for the schedule management system. When everyone involved provides inputs to the schedule management system, the system becomes particularly effective. Owing to the fact that some people derived an immediate advantage of making their schedules public, the schedule management system did not become popular because of its groupware nature [7]. From the advanced research, people in an enterprise used the system more because it encourages users to input data into their schedule [8]. The target audience of this paper is not an enterprise, but rather a community of friends and a family. In other words, we assume the users of our system to be more relaxed than those that might be found in a purely business-oriented environment; hence, the word “casual” has been used in the title of the paper. However, a schedule management system does not provide enough of an effect in a relaxed community, and therefore, the purpose of this system is to motivate users to input data to their schedules continuously.

4 Character Schedule

In this section, we describe the schedule management system called Character schedule.

4.1 Design Policy

The purpose of the system is to provide users with a schedule management tool that they are motivated to use continuously.

1. Use of an avatar in the schedule management system

We incorporated an avatar into the schedule management system. We developed some functions, example given later, using avatars. A user can show other users’ their avatar and can change the outfits of the avatars. The system links the number of schedule items to the motion of the avatars in order to enhance the entertainment value. We expect the entertainment element to encourage users to use the system continuously.

2. Provision using mixi

Our system functions on mixi, a well-known social networking service (SNS) in Japan. This is because our policy to motivate them by adding an activity (using avatar and dressing them) which takes more time. Therefore, we provide our system on SNS that has high the usage frequency in daily life. We believe that social communications can create the chance of the scheduler use.

Character schedule includes functions for the avatar called “dress-up of avatar” and “motion of avatar.” We included the function called “dress-up of avatar” because many existing systems have a similar function, and users are already familiar with it. Moreover, we linked the number of dress-up items to the number of input schedule items. A user can obtain dress-up items by including more items

in their schedule. The motion of an avatar also gives an intuitive indication of the number of items in the schedule. The avatar is a user's representation of the schedule management system, and we believe the appearance and motion of the avatar will influence the entertainment value of the system.

4.2 System Configuration

We developed the Character schedule on mixi using PHP, MySQL, and gadget XML. Mixi is the most well-known SNS in Japan. Character schedule consists of three screens, namely, an avatar edit screen, an avatar plaza screen, and a schedule screen.

1. Avatar edit screen

Figure 1(a) shows the avatar edit screen of the Character schedule. A user can change the appearance of each avatar on the avatar edit screen. The number of dress-up items changes depending on the number of inputted schedule items. When a user inputs the number of inputted schedule items, he/she obtains the avatar items.

2. Avatar plaza screen

Figure 1(b) shows the avatar plaza screen. This is where the user and others can view the avatar. The user can touch an avatar and change its position. If the user clicks on an avatar, he/she obtains detailed information about the user, such as the user name, nickname, the number of the avatar, and the nearest schedule. Table 1 lists the possible motions of an avatar. If a user inputs 10 or more schedule items, an avatar looks busy. We decided on this threshold level of avatar motion based on our preliminary experiment results.

3. Schedule screen

The schedule screen enables the user to input schedule items. Figure 1(c) shows the schedule page. This is also where the user can view a schedule. A fully functional schedule management system consists of various features including layering other user's calendar on top of yours, viewing schedules at different time granularities (day/month/week), etc. We provide the limited functions of a schedule management system for the experiment. The system can only show the schedule of the members by month. The users can check for schedule conflicts using the system.

Table 1. Avatar motion

Number of input of schedule items	Motion of avatar
0	Lying
1~3	Walking
4~8	Walking and running alternately
9 and more	Running and then exhausted gesture

The input situation of the schedule has reflected the schedule for one month.

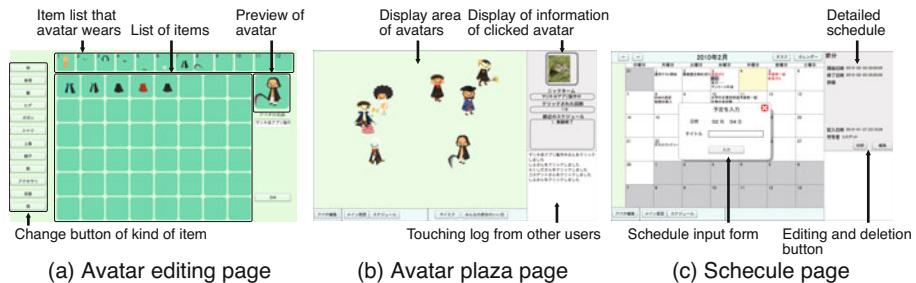


Fig. 1. Screenshots of avatar edit page, avatar plaza page, and schedule page

5 Evaluation Experiment

We evaluated the Casual schedule by performing experiments. The purpose of the experiments is to verify how well the system maintains the user's motivation by means of the avatar and SNS functionality, to continue using the system.

5.1 Verification Items

We verified the following items.

1. What effect does the avatar have on maintaining motivation?
2. What effect does the SNS have on how much the users make use of the system?

5.2 Experiment

Our experiment was performed from 15 January to 29 January 2010, which was approximately two weeks. The subjects were 12 college students who majored in Information Sciences. We instructed the subjects to perform the following tasks.

- (1) We asked the subjects to input three schedule items at first login.
- (2) We showed an instructional movie to the subjects and asked them to read an instruction leaflet.
- (3) We asked the subjects to use the system on a daily basis.
- (4) After the experiments, we carried out a questionnaire survey. We used a five-point Likert scale for evaluation.

5.3 Results of Questionnaire Survey

Table 2 summarizes the results of the questionnaire survey about the avatar aspect of the system. We did not obtain a high evaluation in Table 2(a). Overall, the subjects gave a low evaluation to the avatar items (“I am not interested in the items for avatar.”). We obtained a value of 3.8 in Table 2(b). Here, the subjects gave a high evaluation to the look of the avatar (“I enjoyed the individual-look avatars” and “I enjoyed the dress-up avatars.”).

Table 2. Questionnaire results for the avatar

Question item	Average	Standard deviation
(a) I felt that I wanted to input the schedule items to increase the item of avatar.	3.1	1.11
(b) I felt seeing other users' avatar happy.	3.8	0.37
(c) I felt that I was happy that the motion of avatar changed according to the input items of the schedule.	3.0	1.00
(d) I was able to make my favorite avatar.	3.8	1.01

5-point Likert scale (1: strongly disagree, 2: disagree, 3: neutral, 4: agree, and 5: strongly agree)

We found that users enjoyed seeing other users' avatars. Some subjects responded saying that the motion of the avatar made it easy to see a user's status. However, some answered that "the differences of the motion of avatar are small, so I did not recognize them."

Table 3 summarizes the results of the questionnaire survey about the schedule. We obtained a low evaluation in Table 3(a). Here, the results showed that other users' schedules did not motivate users to input schedule items. The correlation coefficient is 0.65 in Table 4, between Tables 3 (b) and (c). We think that the subjects who answered that another user's schedule was useful tended to input their own schedule more positively.

Table 5 summarizes the results of the questionnaire survey about Character schedule. We obtained a neutral result in Table 5(a) and a high evaluation from Table 5(b). Many comments from users indicated that they used the system because they use the SNS every day. Therefore, we found that providing our system on the SNS has a positive effect on the user.

5.4 Analysis of Operation Log

Figure 2 shows the time for which each user remained on the system per access. In the first three days after the start, users remained on the system for a long time. This was because the subjects needed more time to dress up their avatar and were inexperienced in the system. After four days, the amount of time users remained on the system fluctuated, but differed only slightly.

6 Discussions

6.1 Motivation Maintenance Using Avatar

Table 6 shows the relation between Table 2(a) and the number of items available to each user for their avatar. We found a strong correlation with a correlation coefficient of 0.73. In other words, highly motivated users obtain a large number of dress-up items for the avatar, enabling them to change the avatar more often.

We think that the experiment was conducted with college students which can be a bias. In other words, they are more oriented on video games than on a schedule management system. We need to conduct a real test with businessmen in the future.

Table 3. Questionnaire results for the schedule

Question item	Average	Standard deviation
(a) Because I had seen other users' schedules, I felt that I wanted also to input my schedule.	2.4	1.04
(b) I input my schedule so that other users might see it.	3.3	0.94
(c) It was useful that I browsed other users' schedules.	3.6	1.19
(d) When my schedule was input, I chose scheduling.	3.4	1.44

5-point Likert scale (1: strongly disagree, 2: disagree, 3: neutral, 4: agree, and 5: strongly agree)

Table 4. Relationship between (b) and (c) on Table 3

User ID	u1	u2	u3	u4	u5	u6	u7	u8	u9	u10	u11	u12
Evaluation of Table 3 (b)	4	4	3	5	2	3	4	2	2	4	4	3
Evaluation of Table 3 (c)	5	4	4	5	2	1	4	2	4	4	4	4

Correlation coefficient: 0.65

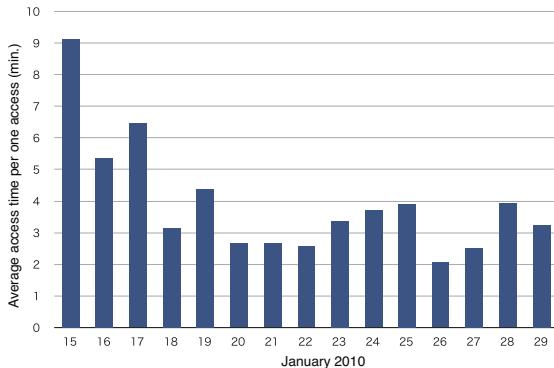
6.2 Effect of Using a Well-Known SNS

We provided our system on a well-known SNS to make the system easier to use. However, we found that we were not able to lower the barriers to use the system from Table 5(a). Some subjects evaluated the system highly, seemingly enjoying using the scheduling facility (“I enjoy inputting my schedule for a change.”). Subjects also appeared to enjoy the fact that they could access the system from the SNS (“I usually use the SNS, and then I use Character schedule.”). We found that making the system available on the SNS made the system easier to use. However, not all subjects were pleased about the connection to the SNS, as they did not use it extensively, and so did not like having to learn two new systems (“I am opposed to using the SNS. I rarely use the SNS. It was too much of a bother” and “I already use another schedule management system. I don’t need another one.”).

Table 5. Questionnaire results for the entire system

Question item	Average	Standard deviation
(a) It was not a burden for me to use the system.	3.2	1.14
(b) When I accessed mixi, I used the system.	4.1	0.95
(c) I want to invite my friend to this system.	3.4	0.86
(d) I want to use the system continuously.	3.3	1.16

5-point Likert scale (1: strongly disagree, 2: disagree, 3: neutral, 4: agree, and 5: strongly agree)

**Fig. 2.** Mean use time per access

6.3 Relation between the Avatar and the Number of Schedule Inputs

Figure 3 shows how well the system achieved the aim of motivating users to keep using the system. On the first day, there were many schedule inputs, because we imposed the task for subjects. After the first day, the number of schedule inputs gradually decreases. The avatar provides an entertainment aspect to the system; however, we believe that we have not provided sufficient effects and options for the avatar.

Table 7 shows the relationship between Table 2(a) and the total number of schedule inputs. The correlation coefficient is 0.68, which is a medium correlation. We found that users who were motivated to obtain items for the avatar tended to input schedule items more often.

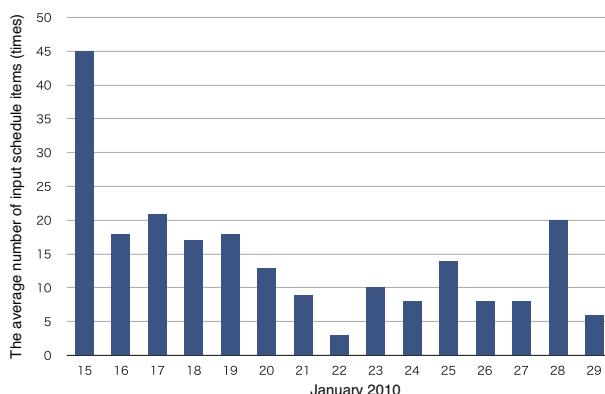
**Fig. 3.** Transition of number of inputs in schedules

Table 6. Relationship between the interest in item acquisition and number of times the attached items changed for the avatar

User ID	u1	u2	u3	u4	u5	u6	u7	u8	u9	u10	u11	u12
Table 2(a)	1	4	4	3	4	2	5	4	2	3	2	3
Number of changing attached item of avatar	27	160	171	63	113	79	377	44	37	81	47	49

Correlation coefficient: 0.73

Table 7. Relationship between total input of each user and interest in the avatar

User ID	u1	u2	u3	u4	u5	u6	u7	u8	u9	u10	u11	u12
Each user of schedule total input and interest of avatar	7	24	12	28	30	22	34	21	11	12	8	9
Evaluation of Table 2(a)	1	4	4	3	4	2	5	4	2	3	2	3

Correlation coefficient: 0.68

Table 8 shows the relationship between the number of schedule inputs by day and the number of dress-up times by day. The correlation coefficient is 0.85, which shows a strong correlation. Therefore, we found that the schedule inputs are strongly associated with the dress-up times.

We split the subjects into two groups: Group A (five subjects) who showed a high evaluation in Table 2(a), and Group B (seven subjects) who showed a low evaluation in Table 2(a). Table 9 summarizes the result of the comparison. We found a significant difference between Group A and Group B in the number of schedule inputs, with Group A inputting more than Group B. Therefore, we found that the schedule inputs are strongly associated with obtaining the dress-up times.

Table 8. Relationship between the number of schedule input items and the number of times the attached items changed for the avatar

	January 15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Number of changing attached item of avatar	352	129	220	61	18	36	8	29	137	37	55	37	41	71	17
Number of schedule input	45	18	21	17	18	13	9	3	10	8	14	8	8	20	6

Correlation coefficient 0.85

Table 9. Comparison of frequency of input in the schedule

	January 15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Group A	3.80	1.60	2.80	1.20	2.80	1.00	1.20	0.40	1.40	0.60	1.80	0.40	1.20	3.00	1.00
Group B	3.57	1.43	0.86	1.00	0.14	1.14	0.14	0.14	0.29	0.57	0.43	1.29	0.00	0.57	0.14

Average: Group A: 1.61, Group B: 0.78

Probability significance is 0.005 (Wilcoxon signed-rank test)

Group A (five subjects) that shows high evaluation (more than and equal 4) in Table 2(a). Group B (seven subjects) that shows low evaluation (less than 4) in Table 2(a).

7 Conclusions

The results of this paper are summarized as follows:

1. We have developed Character schedule including an avatar to motivate users to maintain their schedules. We found that the system is useful for motivating users.
2. From the result of our experiments, we found a strong correlation between the number of schedule inputs and the number of times the attached items changed for avatar.
3. We provided our system on mixi—a well-known SNS in Japan. This helps to encourage access to the system.

We believe that research into motivating users to maintain their schedules is in a very early stage. As a result, we have attempted various trial-and-error methods to develop an effective method for motivating users.

References

1. Linden Lab: Second Life, <http://jp.secondlife.com/>
2. Nicotto Town, <http://www.nicotto.jp/>
3. Kuramoto, I., Kashiwagi, K., Uemura, T., Shibuya, Y., Tsujino, Y.: Weekend battle: an entertainment system for improving workers' motivation. ACM International Conference Proceeding Series, vol. 265, pp. 43–50 (2005)
4. Kuramoto, I., Katayama, T., Shibuya, Y., Tsujino, Y.: A Virtual Aquarium Based on EELF with Subjective Competition for Keeping Workers' Motivation. Journal of Information Processing 50(12), 2807–2818 (2009)
5. Chao, D.: Doom as an interface for process management. In: Conference on Human Factors in Computing Systems, pp. 152–157 (2001)
6. Bernhaupt, R., Boldt, A., Mirlacher, T., Wilfinger, D., Tscheligi, M.: Using emotion in games: emotional flowers. ACM International Conference Proceeding Series, vol. 203, pp. 41–48 (2007)
7. Jonathan, G.: Groupware and social dynamics: eight challenges for developers. Commun. ACM 37(1), 92–105 (1994)
8. Jonathan, G., Leysia, P.: Why groupware succeeds: discretion or mandate? In: Proceedings of the fourth conference on European Conference on Computer-Supported Cooperative Work, pp. 263–278 (1995)

Buyer Coalitions on JADE Platform

Laor Boongasame, Sakon Wathanathamsiri, and Akawuth Pichanachon

Department of Computer Engineering,
Bangkok University,
Bangkok, Thailand

laor.b@bu.ac.th, kon_ls24@hotmail.com, akawuth.pich@gmail.com

Abstract. A variety of existing buyer coalition schemes, but there are a few researches concerning explicit implementation buyer coalition. Additionally, there is no research that implements buyer coalition on major agent platforms such as JADE in split of the success of systems is dependent on the availability of appropriate agent platforms. This paper has conduct on applying JADE framework to address issues in the field of buyer coalition.

Keywords: electronic commerce; coalition formation, buyer coalition. multi-agent, JADE.

1 Introduction

A buyer coalition is a group of buyers who team up to negotiate with sellers for purchasing identical items at volume discount [5]. The reasons that buyer coalitions are increasingly becoming important are 1) buyers can improve their bargaining power to negotiate with sellers in purchasing goods at lower prices [5] and 2) both buyers and sellers will get benefit from a buyer coalition. Buyers can enjoy from purchasing the items in large volume through buyer coalitions if the price of the volume is less than the retail price. On the other hand, sellers get benefit from selling the items at larger volume via buyer coalitions if the cost of the wholesale marketing is less than that of the retail marketing. Many buyer coalition schemes already exist [12,3,4,5,6,7,8,9,10,11,12,13,14,15,16]. However, there are a few researches concerning explicit implementation buyer coalition. Firstly, this research shows use of the test-bed system as a tool for implementation of a real-world buying club [5]. Secondly, the system in the E-Group Buying Market is built as CGI written in Perl. The user interface is written in JavaScript and Html [12]. Although, the success of systems is dependent on the availability of appropriate agent platforms [17], there is no research that implements buyer coalition on major agent platforms such as JADE, Jack intelligent agents, LEAP, FIPA-OS, and ZEUS. A JADE platform is composed of agent containers that can be kept the high performance of a distributed agent system implemented with the Java language and complies with the Foundation for Intelligent Physical Agents (FIPA) standard. It is developed by Telecom Italia (TI). JADE has

been applying in a various and applications such as health care system [18]. This paper has conduct on applying JADE framework to address issues in the field of buyer coalition. It presents a unique approach of using JADE to develop a highly distributed information infrastructure that is able to perform ubiquitous electronic buyer coalition monitoring automatically. The article is organized as follows: Section 2 presents related works. Section 3 provides JADE implementation of BC. Section 4 presents usage scenarios and implementation. Finally, Section 5 conclusion and future works.

2 JADE Implementation of BC

This section presents JADE architecture in subsection 2.1 and BC System architecture in subsection 2.2.

2.1 JADE Architecture

A JADE platform is composed of agent containers that can be distributed over the network and complies with the Foundation for Intelligent Physical Agents (FIPA) standard [17]. Each container or runtime environment can contain several agents. A set of active containers or a platform, there is a special container that must be always active, called main-containner, and all other normal containers have to register with the main-container before they start. The components of main-container are the Agent Management System (AMS) agent and Directory Facilitator (DF) agent. The AMS is the agent who supervises over access to and use of the Agent Platform. Only one AMS will exist in a single platform. The AMS provides white-page and life-cycle service, maintaining a directory of agent identifiers (AID) and agent state. Each agent must register with the AMS in order to get a valid AID or a unique name and also deregister with AMS as it terminates. The DF is the agent that implements the yellow pages service, used by any agents wishing to register its services or search for other available service. The Message Transport System, also called Agent Communication Channel (ACC), is the software component controlling all the exchange of messages within the platform, including messages to/from remote platforms.

JADE complies with the reference architecture of FIPA as Shown in Fig.1

2.2 BC System Architecture

The proposed BC was built on top of JADE, which is suitable to operate in a heterogeneous, networked environment such as the internet to form a buyer coalition . The BC is composed of three types of architectural components, which correspond with human agent in the real world scenario: (1)Buyer Agent, (2)Third-party Agent, and (3)Seller Agent. The overall framework of the proposed BC is depicted in Fig.2

Each component of the architecture and its general activities and purpose within the infrastructure is described below

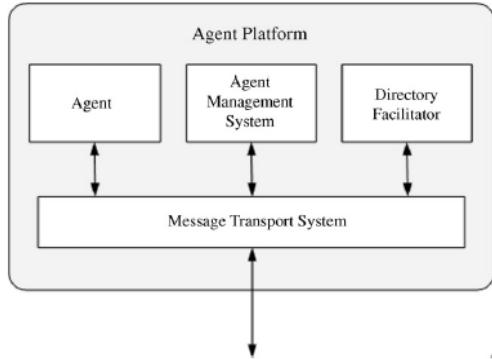


Fig. 1. The reference architecture of a FIPA agent platform

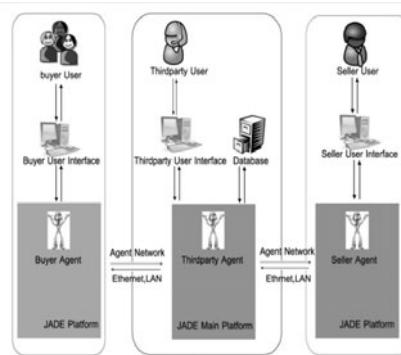


Fig. 2. The BC architecture

Third-party agent. The third-party agent can be considered as a service. It is capable of collecting a buyer coalition data and forming the coalition. The main task of third-party agent is to (1)collect the data from buyers in a coalition during bidding and (2)form the buyer coalition after the bidding is closed and (3)declare the result of forming the coalition automatically to the buyers.

Buyer agent. The buyer agent is regarded as a bridge to interface with third-party agents and buyers. When receiving requests from buyers, it will drive internal services in the platform. The buyer agent is also responsible for the final presentation of results by third-party agent to buyers.

Seller agent. The seller agent is agent used by a seller. It is a computer program that can help a seller to perform their tasks.

Once JADE platform was setup, we can initiate the graphical user interface (GUI) of the BC. Fig. 3 shows the main components in the BC. The main container comprises JADE management agents (e.g., agent management system agent and directory facilitator agent). A container needs to register with the

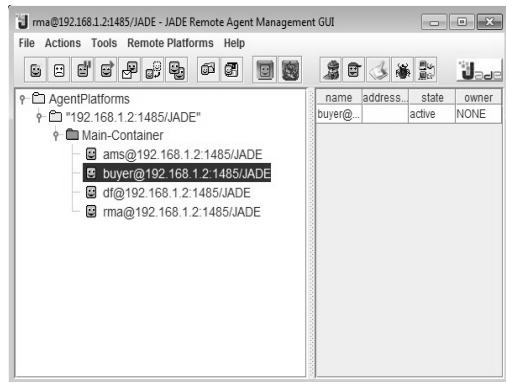


Fig. 3. The main components in the BC

main-container when it gets initiated and displayed in the GUI. As an example shown in Fig. 4, when the Third-party Container get initiated, they register with the main-container and are displayed in the GUI.

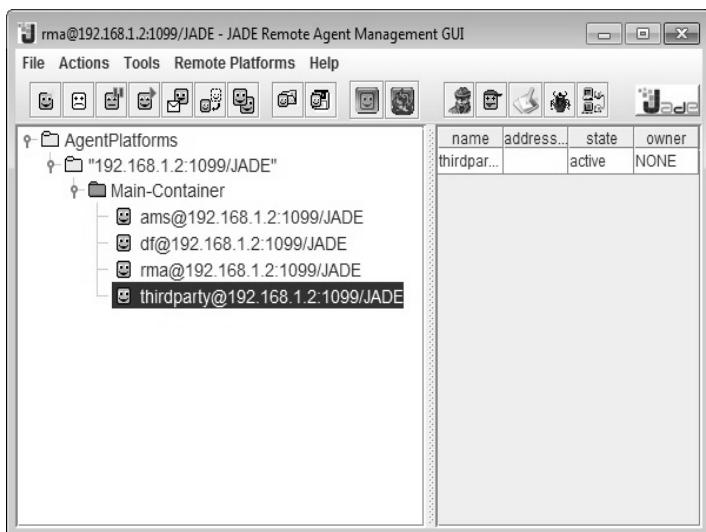


Fig. 4. Containers register with the main-container

3 Usage Scenarios and Implementation

In this section, we describe the implementation scenarios from these two types of users.

3.1 Scenario 1: Third-Partys perspective

While a third-party wants to open buyer coalition, he interacts with the third-party agent and opens buyer coalition through a computer user interface (step 1 of Fig. 5). Upon receiving the order from the third-party, the third-party agent invites buyers to join a coalition via the agent network (step 2 of Fig. 5). The third-party agent will continue delivering message until all the buyers get the invitation. The buyer agents will show all the information via the GUI (step 3 of Fig. 5). The third-party may then switch off his computer and proceed with his daily work. In the meantime, the buyer agents will accept or reject the invitations and send back to the third-party agent (step 4 of Fig. 5). Finally, the third-party agent will show average reservation prices of the buyers who intend to join the coalition via the GUI (step 5 of Fig. 5) when he is back online with his computer .

3.2 Scenario 1: Implementation

Within this scenario, a third-party first login to the system. The third-party will select an item from item lists and then the third-party agent searches the item from databases. The price list of the item is displayed in the middle of the page in title “Item Detail & Price” as illustrated in Fig. 6.

Through the third-party agent interface, the third-party will set deadline of forming a coalition of the selected item and then click “open buyer coalition, send invitation”. The third-party agent will send messages in order to invite all buyers for forming the coalition. Then, average reservation prices of buyers who intend to join the coalition will be displayed in the right of the page in title “Live Update”. During open the buyer coalition, the third-party can cancel the coalition by click “cancel coalition”. In general, the coalition is closed when deadline time is reached. Nevertheless, it can be closed when the third-party click “forming a coalition now” to forming the coalition before deadline time is reached. Finally, when the coalition is closed, the third-party agent will declare

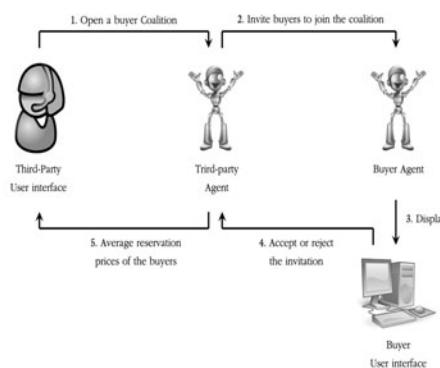


Fig. 5. The usage scenario from a third-partys perspective

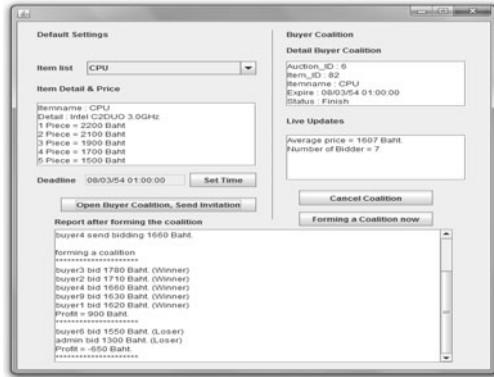


Fig. 6. Third-party agent main window in scenario 1

list of the winners to all buyers and then the third-party agent will show the result in the bottom of the page in title “Report after forming the coalition” via the GUI. The communication among the agents in this scenario is illustrated in Fig. 7.

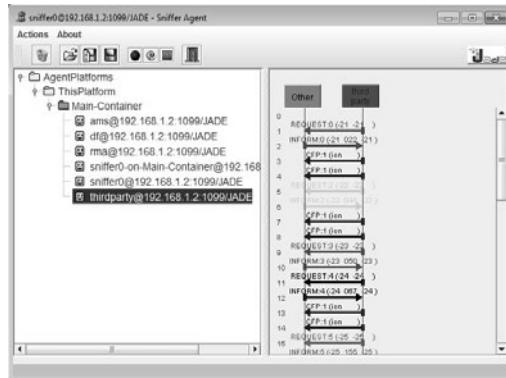


Fig. 7. The communication among agents in the scenario 1

3.3 Scenario 2: Buyers Perspective

While a buyer wants to propose his bid, he interacts with the buyer agent and makes a bid to collect his up-to-date bid data through a computer user interface (step 1 of Fig. 8). Upon receiving the order from the buyer, the buyer agent sends his reservation price to a third-party agent via the agent network (step 2 of Fig. 8). The buyer may then switch off his computer and proceed with his daily work. In the meantime, the third-party agent will surf in the logical agent network to update the information it needs to databases (step 3 of Fig. 8). When the needed information is write/rewrite completely, the third-party agent will

acknowledge (step 4 of Fig. 8) and send the data back to the buyer agent (step 5 of Fig. 8). Finally, the buyer agent will show all the information via the GUI (step 6 of Fig. 8) when he is back online with his computer.

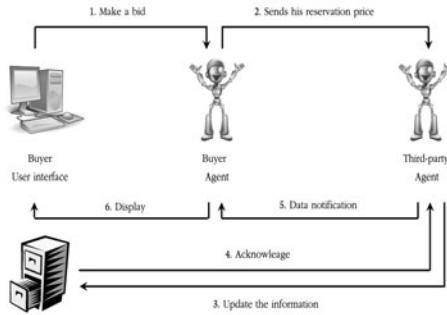


Fig. 8. The usage scenario from a Buyers perspective

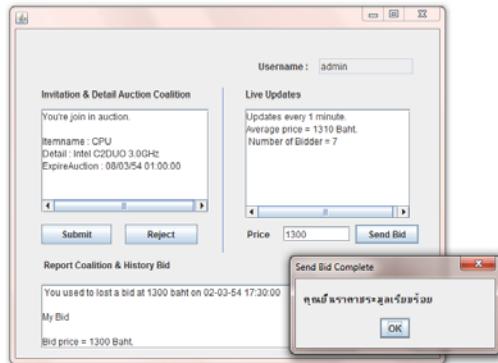


Fig. 9. The buyer agent main window in scenario 2

3.4 Scenario 2: Implementation

Within this scenario, a buyer first login to the system. The buyer will then select to accept or reject invitation of the third-party in order to form a coalition. Then, buyers who intend to join the coalition can insert their reservation prices and pressing “send bid”. Next, a buyer agent is then initiated and dispatched to the network to perform delegated the data. The third-party agent is requested to update the bid of the buyers into the databases. Once the data update is complete, the third-party agent transmits the pertinent data back to buyer agent. Finally, when the coalition is closed, the third-party agent broadcast list of the winners to all buyer agents and then the buyer agents will show the result via the GUI.

4 Conclusions

In this paper, we described development of a multi-agent based distributed information platform BC on top of JADE for buyer coalition application. The proposed BC is expected to reduce the time-consuming task of constant data monitoring of participants and support essential operations such as communication and coordination. The algorithm of forming a coalition in the third-party agent can be enhanced to achieve goals.

References

1. Gurler, U., Oztop, S., Sen, A.: Optimal Bundle Formation and Pricing of two Products with Limited Stock. *International Journal of Production Economics* 118(2), 442–462 (2008)
2. He, L., Ioerger, T.: Combining Bundle Search with Buyer Coalition Formation in Electronic Markets: A Distributed Approach through Explicit Negotiation. *Journal of Electronic Commerce Research and Applications* 4(4), 329–344 (2005)
3. Anand, K.S., Aron, R.: GroupBuying on the Web: A Comparison of Price-Discovery Mechanisms. *Journal of Management Science* 49(11), 1546–1562 (2003)
4. Laor, B., Leung, H.F., Boonjing, V., Dickson, K.W.: Forming Buyer Coalitions with Bundles of Items. In: Nguyen, N.T., Hakansson, A., Hartung, R., Howlett, R., Jain, L.C. (eds.) *KES-AMSTA 2009. LNCS*, vol. 5559, pp. 714–723. Springer, Heidelberg (2009)
5. Tsvetovat, M., Sycara, K., Chen, Y., Ying, J.: Customer coalitions in electronic markets. In: Dignum, F.P.M., Cortés, U. (eds.) *AMEC 2000. LNCS (LNAI)*, vol. 2003, pp. 121–138. Springer, Heidelberg (2001)
6. Hyodo, M., Matsuo, T., Ito, T.: An Optimal Coalition Formation among Buyer Agents based on a Genetic Algorithm. In: Chung, P.W.H., Hinde, C.J., Ali, M. (eds.) *IEA/AIE 2003. LNCS (LNAI)*, vol. 2718, pp. 151–157. Springer, Heidelberg (2003)
7. Indrawan, M., Kijthaweesinpoon, T., Srinivasan, B., Sajeev, A.: Coalition Formation Protocol for E-Commernce. In: Proceedings of the International Conference on Intelligent Sensing and Information Processing, Chennai, India, pp. 403–407 (2004)
8. Kraus, S., Shehory, O., Tasse, G.: Coalition Formation with Uncertain Heterogeneous Information. In: Proceedings of the Second international Joint Conference on Autonomous Agents and Multiagent Systems, Victoria, Australia, pp. 1–8 (2003)
9. Kraus, S., Shehory, O., Taase, G.: The Advantages of Compromising in Coalition Formation with Incomplete Information. In: Proceedings of the Third international Joint Conference on Autonomous Agents and Multiagent Systems, NewYork, USA, pp. 588–595 (2004)
10. Li, C., Sycara, K.: Algorithm for Combinatorial Coalition Formation and Payo Division in an Electronic Market place. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, Bologna, Italy, pp. 120–127 (2002)
11. Li, C., Rajan, U., Chawla, S., Sycara, K.: Mechanisms for Coalition Formation and Cost Sharing in an Electronic Marketplace. In: Proceedings of the 5th International Conference on Electronic Commerce, Pennsylvania, USA, pp. 68–77 (2003)

12. Matsuo, T., Ito, T., Shintani, T.: A Buyers Integration Support System in Group Buying. In: Proceedings of the IEEE international Conference on E-Commerce Technology, Washington, DC, USA, pp. 111–118 (2004)
13. Matsuo, T., Ito, T., Shintani, T.: A Volume Discount-based Allocation Mechanism in Group Buying. In: Proceedings of the 2005 International Workshop on Data Engineering Issues in E-Commerce, Tokyo, Japan, pp. 59–67 (2005)
14. Yamamoto, J., Sycara, K.: A Stable and Efficient Buyer Coalition Formation Scheme for E-Marketplaces. In: Proceedings of the 5th International Conference on Autonomous Agents, Montreal, Canada, pp. 576–583 (2001)
15. Chen, J., Chen, X., Song, X.: Bidders Strategy Under Group-Buying Auction on the Internet. *IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans* 32(6), 680–690 (2002)
16. Chen, J., Chen, X., Kauffman, R.J., Song, X.: Should we collude? Analyzing the benefits of bidder cooperation in online group-buying auctions. *Electronic Commerce Research and Applications* 8(4), 191–202 (2009)
17. Bellifemine, F., Caire, G., Greenwood, D.: Developing Multi-agent System with JADE. John Wiley & Sons, Ltd., Chichester (2007)
18. Su, C., Wu, C.: JADE implemented mobile multi-agent based, distributed information platform for pervasive health care monitoring. *Applied Soft Computing* 11, 315–325 (2011)
19. Endsuleit, R., Calmet J.: A Security Analysison JADE(-S). In: Proceedings of NORDSEC, Estonia, V.3.2 (2005)

Emotional Agents in a Social Strategic Game

Weiqin Chen, Christoph Carlson, and Mathias Hellevang

Department of Information Science and Media Studies, University of Bergen,
P.O.Box 7802, N-5020 Bergen, Norway

Abstract. Existing research shows that emotions affect the decision making of the agent and make it appear more human-like. In game industry, emotions of players and non-players characters have not been paid enough attention. This paper presents the development of an Emotion Module in StateCraft, a software version of the social strategy board game Diplomacy, with aims to add emotion to agents in order to make the agent more believable and fun to play with. Based on the OCC-model we conducted interviews with players in order to identify what emotions are experienced in the game. These emotions are then mapped onto the OCC-model and implemented in the agent. We conducted simulations as well as player test to study how the emotions affect the agent performance and player experience.

1 Introduction

Emotions have shown to be an important part of human intelligence which plays an important role in decision-making. According to Ortony et al. [8], emotions are valenced reactions to events, agents, or objects, with their particular nature being determined by the way in which the eliciting situation is construed. Most emotion theorists agree that some emotions are more basic than others, often called primary or fundamental emotions. However, they tend to disagree on which emotions are basic, why they are basic emotions or how many basic emotions there are [7] [4]. Despite of the lack of agreement, emotions are considered an important part of human intelligence. In fact, research has shown that emotions can be successfully implemented in agents. The results indicate that emotions can improve both the performance of an agent [6], as well as the believability of the agent, where believability refers to the agent providing the illusion of life [1].

From an engineering perspective, emotions can help agents make better decisions, hence improve performance. Maria and Zitar [6] showed that an agent with emotions performed better than a regular agent in a benchmark problem. Their research indicated that emotions can enhance performance if used correctly. In computer games, emotions are used to create a more human-like opponent, making it more fun to play with. The Ortony Clore Collins-model (OCC) is a widely accepted model to synthesise emotions in agents [6] [3] [1]. It focuses on what contribution cognition makes to emotion, and devotes less time to other important aspects such as facial expressions or behavioural components. Nor does it focus on how different emotions interact with each other. The OCC-model puts

emotions into 22 categories. Depending on if they are appraisals to events, actions of agents or objects, it gives these emotions different emotion words. In the OCC-model, emotion was described as a valenced reaction to an event, agent or an object. Valenced means that the emotion has to get a positive or negative reaction, excluding neutral emotions such as surprise. The emotion's particular nature is being determined by the construal of the eliciting situation. Events are considered things that happen, and the agent's reaction depends on its goals. Actions of agents can be approved or disapproved depending on the agent's set of standards. If it is another agent who performs the action, it can give rise to the emotions admiration and reproach. If it is the agent itself who carries out the action, the emotion pride or shame might occur. Objects can be disliked or liked based on the agent's attitudes towards the object. An object can also be another agent. An important component in the OCC-model is that an emotion's intensity has to be above a certain threshold value. If the value is below the threshold value, the emotion will not be experienced by the person.

In this research we developed an Emotion Module in StateCraft, a software version of the strategy board game Diplomacy. We aim at adding emotion to agents in order to make the agent more believable and fun to play with. We have chosen to use the OCC-model because it is computation-oriented and used in different projects for synthesizing emotions in agents.

The rest of the paper is organised as follows. In section 2 we introduce the Diplomacy game and the software version of it. Section 3 describes the emotions we identified in Diplomacy based on user study and the OCC-model. In section 4 we present the implementation of the emotions in the EmotionSynthesizer. Section 5 describes the user test and simulations we have conducted to study how the emotions affect the performance of the agent and player experience. Finally we conclude the paper in section 6.

2 Diplomacy and StateCraft

Diplomacy is a strategy-based social board game. It simulates the First World War when seven nations fought for domination over Europe. The seven nations include England, France, Germany, Russia, Italy, Austria-Hungary, and Turkey. The board is a map of Europe (showing political boundaries as they existed in 1914) divided into 75 regions of which 34 contain supply centres. For each supply centre a player controls, she or he can build and maintain an army or a fleet on the board. If one of the players controls 18 supply centres, this player has won the game. The game mechanics are relatively simple. Only one unit may occupy a region at any time. There is no chance involved. If the forces are equal in strength, it results in standoff and the units remain in their original positions. Initially each country is roughly equal in strength, thus it is very difficult to gain territory - except by forming alliance and then attacking. Negotiation for forming alliances is a very important part of the game, because numerical superiority is crucial. Secret negotiations and secret agreements are explicitly encouraged, but no agreements of any kind are enforced.

Each game turn begins with a negotiation period, and after this period players secretly write orders for each unit they control. The orders are then revealed simultaneously, possible conflicts are resolved and the next turn can commence.

StateCraft is a software version of the Diplomacy, developed by Krzywinski, et al. [5]. It is an online multiplayer turn-based strategy game where each of the seven countries can be played by either a human player or an agent (Fig. 1). A three-layered agent architecture is developed which includes an operational layer, a tactical layer and a strategic layer. These layers handle three main tasks when playing Diplomacy respectively, monitoring the game board, planning moves, and engaging in diplomatic negotiations. The operational and tactical layers are invoked once each turn, acting on the new game state that resulted from the previous round and the strategic layer is active throughout the whole game session. Thus the agent is driven by the periodical updates of the game state, while still maintaining continuous diplomatic interaction with the opponents.

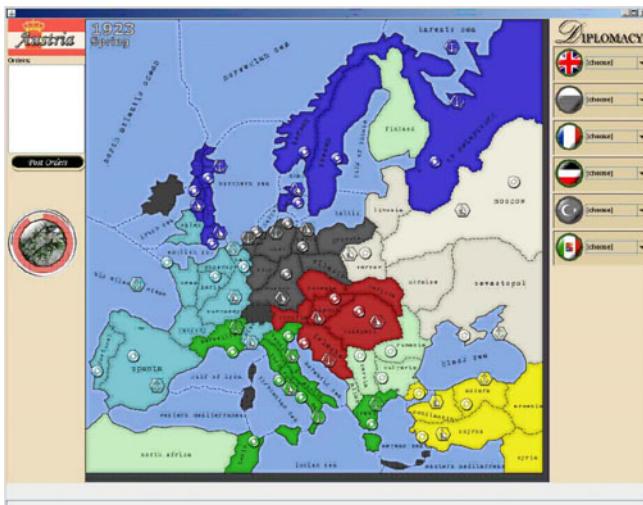


Fig. 1. Game interface for players

The operational layer is a reactive layer and is triggered at the start of each round. It monitors the game board and discovers all possible and legal moves for each unit based on the game state. The tactical layer combines operations for each unit into a set of operations, a tactic. Each tactic contains thus one operation for each of the agents units. Each tactic has two values, potential value and factual value. While potential value represents the value of a tactic regardless of the other players moves, the factual value represents the tactics tactical value combined with its chance for success. If a tactic has a very high potential value, but is considered impossible to achieve, its factual value will be much lower, while a high certainty for success will give similar values for potential and factual value. The strategic layer is responsible for communicating with the other players, and based on this diplomatic activity and the weighted tactics from the previous two layers, selects the appropriate tactic for the current round. This layer is

organized according to the Subsumption Architecture [2] and in StateCraft it consists of four relatively simple modules – Choose Tactic, AnswerSupportRequest, SupportSuggester and Relationship (Fig. 2). The result is a more flexible structure, as the modules impact each other in a non-linear manner and new modules can be added without breaking the former functionality.

3 Emotions in Diplomacy

To synthesise emotions in the agent, we considered two emotional models. They were the OCC-model [8] and the three-layered architecture [9]. The OCC-model was selected because simplified versions of the OCC-model have been implemented into several projects, while the three-layered architecture lacks implementation specific details.

3.1 Identifying Emotions in Diplomacy

In order to study what emotions are experienced and how the emotions influence the decision-making when playing Diplomacy, we invited seven players to play the game and conducted interview afterward. During the interview, the players were asked to describe their emotions and how the emotions affected their choices.

The result from the interview was summarized. The most frequently experienced emotions were joy, loyalty, guilt, fear, anger, shame, relief and disappointed. Relief and disappointment lead to the same outcome as joy and anger, shame overlaps with guilt. Joy, fear, anger and admiration are categorized in the OCC-model. According to this model, joy is considered as the reaction to an event which was construed as desirable by the person. Fear is the prospect of a situation which would be undesirable for that person. Anger is the combination of distress because of an undesirable event and the feeling of reproach towards a person who did something one does not approve of. Admiration is the reaction to another agent’s action which one approves of. Loyalty is not defined in the OCC-model as it is not a valenced reaction to events, agents or objects. However, the situations where the players reported that they experienced loyalty seem to be when they approved of another player’s actions. In the OCC-model, admiration occurs when one approves another agent’s actions. Ortony et al. [8] specified that the meaning of the words used to describe emotions in the OCC-model were not necessarily equivalent to the meaning of the words in spoken English. Therefore, we could assume that what the players actually meant was the emotion that is structured as admiration in the OCC-model, since an effect of admiration often will be loyalty to the admirer. Guilt is not categorised by the OCC-model either. Based on the players statements, we could assume that guilt is a sub-category of the emotion shame. Guilt involves disapproving of one’s own action towards another person, while also feeling sorry for that other person. This could be structured as a compound emotion in the OCC-model. For simplicity reason, we considered guilt a sub-category of shame.

3.2 Emotion Intensity

The intensity of an emotion is represented by a numeric value between 0 and 100, where all emotions start at a default value of 0. For an emotion to affect the agent's decisions it needs to exceed the threshold set for the particular emotion. All emotions have a different intensity directed towards each player, except from joy, which only has a general intensity. The emotions joy and admiration also have the possibility of negative values down to -100. A joy value of -100 represents the opposite of joy, which in the OCC-model is distress. A negative admiration value represents reproach. Alone, distress or reproach does not affect the decisions made by the agent, but together they form the emotion anger. The anger's intensity is calculated by calculating the square root of the absolute value of distress multiplied with reproach, provided that both of these emotions exceed the negative threshold.

To ensure that the agent has a clearly defined emotional state, the agent can only have one emotion towards each country at the same time. The strongest emotion will suppress the other emotions. For instance, when the agent playing Germany feels very angry and a little afraid of the agent playing Austria, since anger is the greatest in intensity, it will suppress the fear. The exception is joy, since joy is general and not directed towards a particular country. However, if fear or guilt is the strongest emotion towards a country, the intensity of joy will decrease. Furthermore, it is impossible to experience joy and anger at the same time, since anger depends on a negative joy value.

4 Implementation of EmotionSynthesizer

In this section we present the implementation and integration of the Emotion-Synthesizer in StateCraft. The EmotionSynthesizer is based on the events that cause the changes in emotion intensity and how these emotions affect the decisions and actions of the agent. The emotions implemented are joy, admiration/reproach, anger, fear and guilt.

Given that the Emotion module is an addition to the agent in StateCraft, the whole module has been implemented in the emotions package in the Strategic layer of the agent in StateCraft. Since the Strategic layer uses an architecture similar to the Subsumption system, using sensors to look for changes in the environment and actuators to act on the changes from the sensors, the Emotion module receives input through an input line from GameStateSensor, the sensor listening for new game states from the server, and MessageSensor, the sensor listening for new SupportRequestMessages and AnswerSupportRequestMessages. Then it performs its actions by suppressing the input to ChooseTactic, the module responsible for choosing tactics. Additionally, it inhibits the output from the AnswerSupport module. Fig. 2 depicts the Emotion module as part of StateCraft's Strategic layer.

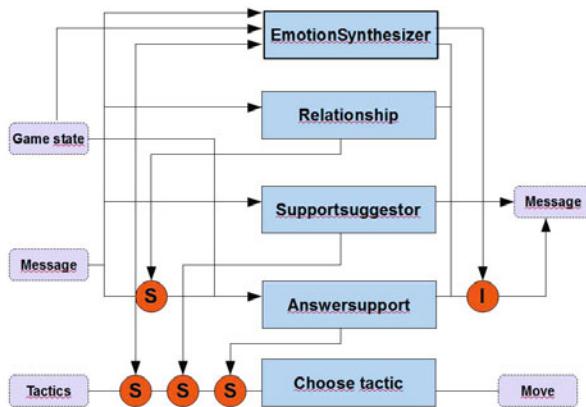


Fig. 2. Emotion module integrated in the strategic layer in StateCraft architecture

4.1 Joy

Joy is the positive reaction to events as specified by the OCC-model. The event that increases the intensity of joy is when the agent gains a province. The intensity depends on the desirability of gaining the province. Gaining a province containing a supply centre will be more desirable for the agent, and therefore increase the joy more than gaining a province without a supply centre. In the emotion module, distress is modelled as the joy emotion with a negative intensity because joy and distress are mutually exclusive. The events that decrease joy are: 1) The agent loses a province. The desirability of keeping the province influences the intensity; 2) The agent asks for support, and the opponent agrees to perform the support, but for some reason avoids or neglects to perform the support; 3) An opponent asks the agent for support and the agent performs the support order, but the opponent does not perform the move order of which they asked support; 4) The agent's strongest feeling towards another player is guilt or fear; 5) The agent is outnumbered by its opponents.

4.2 Admiration

Admiration is the approving reaction to actions by other agents, while reproach is the disapproving reaction. Actions made by other agents of which the agent approves are: 1) An opponent agrees to support the agent and keeps the deal by supporting the agent's move order; 2) An opponent with adjacent units does not attack the agent.

Reproach is represented as admiration with a negative value in the emotion module. The actions which decrease admiration are: 1) The agent asks an opponent for support and the opponent accepts. Despite this, the opponent does not perform the support move; 2) The agent is attacked by an opponent. The intensity of the emotion depends on what the agent expects of the opponent. If

the opponent and the agent has a friendly relationship, the admiration towards the opponent will decrease more than, for example, if they were at war.

Admiration towards opponents will decrease the chance of the agent attacking them. In addition, the agent will more likely perform the support moves it has promised towards the opponents it admire.

4.3 Anger

Anger is the combination of reproach and distress, meaning that when admiration and joy decreases, anger increases. The situations that decrease admiration and joy cause anger to increase (see 4.1 and 4.2). Anger towards opponents will increase the chance of the agent attacking the opponents. It will also decrease the chance of the agent supporting them.

4.4 Fear

Fear is the negative expectation to an upcoming event. The negative events which the agents are likely to fear in StateCraft are to be attacked by an agent, and to be lied to by a player who agreed to support the agent. For the sake of simplicity, we have chosen to focus on the fear of being attacked by a mighty neighbour. The intensity is calculated based on two factors: *Probability*. The more adjacent provinces the opponent has to the agent's provinces, the more likely that the opponent will attack; *Damage*. A more powerful opponent can do more damage to the agent than a less powerful opponent.

Fear towards opponents will decrease the chance of the agent attacking them. It will make the agent more defensive.

4.5 Guilt

Guilt is the disapproving reaction to one's own action combined with feeling sorry for the agent(s) which the action affected in an undesirable way. The events which increase guilt are: 1)The agent attacks an opponent to whom it has a friendly or neutral relationship; 2) The agent agrees to support another player, but avoids performing the support operation. The intensity is also dependent on the relationship to the other player. If the agent performs a support towards a player towards whom it feels guilty, the guilt intensity will be reset to zero.

Guilt towards opponents will decrease the chance of the agent attacking them and make the agent less reluctant to support them.

5 Evaluation

We have conducted evaluations to study how the emotions affect the performance of the agent and player experience in StateCraft. Simulations using various configurations of the Emotion module and statistical analysis were used to study how emotions affect the performance in terms of supply centres. User test was conducted to understand the player experience, e.g. whether they can identify the different emotions of the agents and whether they think it is fun to play against agents with emotions.

5.1 Simulations

We ran simulations of the game from 1901 to 1911 all 7 countries were played by agents with 9 different configurations: 1) all agents with emotions, 50 simulations; 2) no agents with emotions, 50 simulations; 3) one of the 7 countries is played by an agent with emotion and all others are played by agents without emotions, 30 simulations each. The data used for analysis was mainly the number of supply centres at the end of the game (Spring 1911).

Table I shows the results from the All emo and No emo simulations. Mean contains the average number of supply centres based on the 50 simulations. Victories contains the number of games where the specific country ended with the most supply centres. Occasionally, more than one country "win", for example when both Germany and France have 9 supply centres and all the other countries have 8 or fewer. Extinctions contains the number of times where the specific country ended the game with 0 supply centres. As shown in Table I, there are small differences between the simulations in No emo and All emo. One of the biggest differences is that Russia, despite having the lowest average, wins as much as 8 games, compared to only winning 2 of 50 games as a regular agent. The beginning seems to be very critical for Russia. If Russia is too aggressive and leave its borders too open, Germany and Turkey takes advantage of this and occupies Russia's home provinces. Russia wins the two games where Germany is exterminated. Italy seems to behave quite similar in each round, that is, they only move down to Tunisia while keeping their original provinces. England also has a rather low standard deviation, and the common denominator for England and Italy is that their provinces are mostly surrounded by water, hence more difficult for opponents to take their centres. This can explain why both Italy and England never become extinct in neither No emo nor All emo.

Table 1. Results from No emo simulations

Country	No emo				All emo			
	Mean	Std dev	Victories	Extinctions	Mean	Std dev	Victories	Extinctions
Germany	7.10	2.978	27	0	7.22	3.388	23	2
Turkey	6.44	2.215	13	1	6.40	1.796	14	0
England	4.80	1.784	4	0	4.44	1.704	5	0
Austria	4.42	2.45	7	3	4.00	2.785	7	2
France	4.38	2.39	4.5	5	4.96	2.222	2	2
Italy	4.16	1.267	0	0	4.16	1.754	2	0
Russia	2.54	2.27	2	6	2.70	2.384	8	8

Differences for individual countries. To study the effects of the Emotion module for a specific country, we compared two data sets where the only difference was the country was running the Emotion module in one of the data sets, while all the other countries were controlled by regular agents. This enabled us to compare the number of supply centres gained by for example Germany with emotions with the number of supply centres gained without emotions. The

results from comparing 30 simulations with/without emotions for each specific country show that the emotional agents win fewer times than the regular agents (11.9% to 16.47% of the games) and they become extinct more often (6.19% to 4.29% of the games).

5.2 User Test

The participants consisted of five males and one female, aged between 23 and 31 years. Four of them had prior experience with Diplomacy, while two had never played the game before, but all of the players had previous experience with other computer games. The inexperienced players were given a brief introduction to the rules and game mechanics of Diplomacy. The participants played Austria in all games, because of Austria's geographic position, with many neighbouring countries. Each participant played 3 games of StateCraft to 1905. Game 0: against a mixture of emotional agents and regular agents and were asked to identify which emotions the agents were feeling towards them; Game A: against all emotional agents; Game B against all regular agents. The sequence of Game A and B were mixed to prevent the effect from that the players become better at the game. After playing through Game A and B they were asked if they observed any differences between the two games, and if so, which differences they found and if they thought one of the games were more fun than the other.

When asked to identify the different emotions after Game 0, only two players managed to pick an emotion that the agent actually felt towards them. By this, we can conclude that players are unable to identify the agents' emotions, at least not on such a short time frame. Only one inexperienced user and one experienced user managed to identify an emotion. It is important to point out that the users do not know how emotions would be expressed through the actions of the agent. In addition, an agent's aggression caused by a high joy intensity may be misinterpreted by the users as the aggression caused by a high anger intensity.

After playing Game A and Game B, four of the players thought that it was more fun to play against emotional agents, one thought it was more fun to play against the regular agents and one did not notice any difference between the two games. Several of the participants mentioned that the game against emotional agents were more fun since they performed better themselves, and therefore the game appealed to their feeling of mastering. This is also supported by the fact that the only player who performed better against the regular agents is the same person that found it more fun to play against the regular agents. One player stated, *"Game A was more fun. Simply because I got 12 centres, but it was the second game that was most challenging. But since I lost in the end, it was less fun"*. When asked if they observed any differences between the two games, all participants except one thought there was a difference.

On average, the players end up with an average of 2.5 supply centres when they play against regular agents compared to an average of 5.5 supply centres when they play against emotional agents. This corresponds with the results from the simulations where Germany and Russia performed far worse with emotions.

Austria, which is the country that the users played, is a neighbour to both Germany and Russia, and this can explain why most of the players performed better against the emotional agents.

6 Conclusion

In this paper we presented the implementation of Emotion Module based on OCC-model and results from player interview. The results from the data simulations show that some countries perform worse with emotions. The countries that perform worse are the countries which are better off using a more conservative approach. However, not all countries are punished for using emotions, and some countries even perform better with emotions. The common denominator for these countries is that their home provinces are hard to infiltrate for opposing countries. We were able to determine that it is very difficult for a human player to identify which emotions the agents were feeling in such a short amount of time. But even if the players did not manage to identify the agents' emotions, it appears that most of the participants thought it was more fun to play against agents with emotions than agents without emotions.

References

1. Bates, J.: The role of emotion in believable agents. *Communications of the ACM* 37(7), 122–125 (1994)
2. Brooks, R.: Elephants don't play chess. *Robotics and Autonomous Systems* 6(1&2), 3–15 (1990)
3. El-Nasr, M., Skubic, M.: A fuzzy emotional agent for decision-making in a mobile robot. In: *IEEE World Congress on Computational Intelligence*, vol. 1, pp. 135–140. IEEE, Los Alamitos (1998)
4. Frijda, N.: Comment on Oatley and Johnson-Laird's Towards a Cognitive Theory of Emotions. *Cognition & Emotion* 1(1), 51–58 (1987)
5. Krzywinski, A., Chen, W., Helgesen, A.: Agent architecture in social games – the implementation of subsumption architecture in diplomacy. In: Darken, C., Mateas, M. (eds.) *Proceedings of AIIDE 2008*. The AAAI Press, Menlo Park (2008)
6. Maria, K.A., Zitar, R.A.: Emotional agents: A modeling and an application. *Information and Software Technology* 49(7), 695–716 (2007)
7. Ortony, A., Turner, T.J.: What's basic about basic emotions? *Psychological Review* 97(3), 315–331 (1990)
8. Ortony, A., Clore, G.L., Collins, A.: *The cognitive structure of emotions*. Cambridge University Press, Cambridge (1988)
9. Sloman, A.: Architectural requirements for human-like agents both natural and artificial (what sorts of machines can love?). In: *Human Cognition and Social Agent Technology (Advances in Consciousness Research)*. John Benjamins Publishing Co., Amsterdam (2000)

Modelling Multimodal 3D Virtual Environments with Asynchronous Multi-Agent Abstract State Machine

Fabio De Felice, Alessandro Bianchi, and Fabio Abbattista

Computer Science Department “Aldo Moro” University Bari, Italy
`{fabio.defelice,bianchi,fabio}@di.uniba.it`

Abstract. Virtual Environments can be considered as asynchronous distributed systems with static and highly dynamic aspects. Despite a number of available design tools, the dynamic aspects, behaviour and interaction, are mainly designed and developed with ad hoc solutions utilizing high-level programming or scripting languages without any engineered design procedure. This leads to neither reusable nor readable solutions letting design of dynamic in Virtual Environments still a complex and time-consuming task. In this work the Asynchronous Multi-Agent Abstract State Machine model is analyzed. Its suitability to the design of dynamic aspects of Virtual Environments is evaluated in order to put the basis to the development of a design methodology unifying the intuitiveness of Agent Based Modeling with the Abstract State Machine theoretical foundations and well defined methodology.

Keywords: Design engineering, Human computer Interaction, Intelligent agent, Modeling, Virtual reality.

1 Introduction

Virtual Environments (VEs) are characterized by dynamic and static aspects. The dynamic aspect of a VE involves all the possible behaviours and interactions associated with virtual entities. The static aspect involves all the information about geometries and appearances of the virtual entities. In this work we propose an approach to design dynamic aspects of VEs by means of Abstract State Machines (ASM) [1] and Agent Based Modelling (ABM) [2]. In particular it is considered multimodal interaction between the user and VEs by means of haptic [3] and acoustic devices.

The design of both aspects of a VE is a very time consuming task. Despite different tools are available today for a more intuitive design approach to static scene, such as Unreal or Google Sketch up, behaviours and interactions are usually implemented with languages or scripting languages. This leads to complex and not reusable code. Model-based approaches exist that allow the specification of dynamics in a more intuitive way. However complex dynamics often results in very complex models [4].

The approach presented in this work exploits the intuitiveness of the ABM and the theoretical foundation and flexibility of ASM in describing complex dynamic systems, to specify behaviours and interactions in a unified and integrated procedure. VE can be considered as an asynchronous distributed system. It is distributed because

contains different types of virtual entities acting and interacting in their neighbourhood in a multimodal way with sight, hearing, touch etc. In particular we consider user's Avatar as a special case of virtual agent. It is asynchronous because all the entities do not share a unique clock coordinating their actions, but each of them follows behaviours governed by its own internal logic or by human user behaviour. This view allows designers to focus on interaction expressivity and allows more flexibility in Avatar behaviour design. We use Asynchronous Multi-Agent ASM (async-ASM) to capture the whole model semantic with a unique formal and rigorous solution.

The work is organized as follows: in the next section general trends in designing dynamics in VE are presented. In section 3 the intended VE conceptual model is given to better describe which are the interacting actors. Some elements of ASM are given in section 4 and section 5 presents a multimodal interactive agent and how it can be modelled with ASM. In section 6 the async-ASM used to model the VE dynamics is reported, while in section 7 conclusions are drawn.

2 Related Work

Many works have been proposed to model VE dynamic through Petri nets based formalisms. In modelling behaviours new models such as High Level Petri Nets [5], Flownet [6] and Behaviour Transition Networks [7] adds capabilities to the basic formalism such as constructs for describing continuous processes. The main disadvantages of these approaches are intrinsic to Petri nets. In [8] it has been shown that Petri nets are not appropriate for formalizing exceptions, streaming and traverse-to-completion. Moreover, according to [9], the lack of a unified Petri-net formalism, integrating the different variants used to map different concepts, represents a serious obstacle for related approaches.

For what concerns multimodal interaction, in [10] a graphical notation, called NiMMiT, both based on state and data driven model is presented. It is particularly focused on the multimodal interaction design. A data driven approach, called InTml, can be found in [11]. This is a markup based description language for multimodal interaction that exploits the X3D syntax adding nodes to specify input/output devices and interaction techniques. These approaches focus on multimodal user interaction and do not provide tools to model multimodal generic entities' interaction.

An approach allowing designing these components in a unified and integrated way is the ABM. With ABM, VEs can be seen as multi-agent systems, in which one, or more, virtual actor can be controlled by a given agent. In [12] taxonomy of agent complexity is presented based on behavioural and implementation complexities. In particular, implementation complexity is defined even based on the degree of social interaction an agent is capable of. In [13] the concept of smart object is introduced. Smart objects are virtual entities controlled by agents that make explicit to the user the semantic of the object itself. Dobrowolsky [14] describes how a given agent-based architecture can be programmed as an ASM to establish a link between well founded notions of multi-agent systems and a systematic design and implementation procedure.

In the VE's context, Mi and Chen [15] proposed agent architecture for user's avatar in the context of Collaborative Virtual Environments in order to enrich the interaction among users. The architecture allows a semi-autonomous Avatar's agent

intelligent behaviour that aims at make some autonomous decisions without overloading the user with too many controls. Usage of ASM in 3D VEs can be found in [16] to describe the dynamic of the 3D simulation, resulting in a concise and unambiguous high-level formal simulator description.

The use of ASM formalism is justified by different point of views. Several similarities exist between Petri nets and ASM and [17] shows that the run in a Petri net can be expressed by ASM rules, in particular it emphasizes the capability of asynchronous distributed ASMs in modelling the distributed nature of Petri nets. The capability of ASM in describing agent based models allows our approach to take into account the entire system in both its static and dynamic aspects, through a set ASMs implementing an asynchronous distributed ASM. Resuming, we applied ASM approach considering the advantages it provides under three different points of view. When the model expressivity is considered, a rich literature (e.g. [8][18]) shows that ASMs have excellent capabilities to capture the behavioural semantics of complex dynamic systems, like multimodal VEs, where several different processes occur, often with the need to properly model exceptions, streaming and traverse-to-completion. Secondly, considering software engineering development issues it is worth noting that starting from the ASM formalism a system development process has been defined and successfully applied in several complex domains, e.g. telecommunication, programming languages, control systems, and so on [19]. Finally, considering the implementation point of view, the lack of specific environments for building executable code starting from Petri nets models is overcome by using an ASM-based approach thanks to tools like AsmL [19] and CoreASM [20].

3 The Virtual Environment Model

As mentioned before a given VE can be divided into static and dynamic aspects. Statically a virtual object can be formally defined as follows:

$$\text{VO} = \langle G, A, P, H, S \rangle \quad (1)$$

Where, G is the geometry of the object and it is usually defined by geometric primitives or obtained by directly specifying the 3D points cloud and its triangulation. The appearance attribute A describes the shape in terms of colours and textures, and how it reacts to light points in the scenes as formalized in the Open Inventor standard [21]. The attributes in P store information about the current position. Other attributes are the H haptic material attribute describing how the object must be felt haptically (stiffness, dumping, static and dynamic friction) and the S attribute describing sounds eventually associated to the object. In a multimodal context there could be different types of behaviour, as much as the number and type of object interactions. Dynamic behaviour describes how the entity moves; visual behaviour refers to the ability of some animals to change the appearance of their skin for camouflage or for courting, acoustic behaviour allows to play sounds effects or TextToSpeech; haptic behaviour allows to express haptic effects such as vibration, attraction and arbitrary force fields.

Interaction is a component strictly related to behaviours because it defines the activation sequence of particular event-driven behaviours. An interaction is considered as a bidirectional flow of communication involving two or more entities: the entity E_1 ,

call it Source, with the behaviour b_1 , interacts with the entity E_2 , call it Target, which in turns activates the behaviour b_2 . This last behaviour can induce behaviour activation in E_1 or in another entity E_3 , or in nobody at all. The sequence of these behaviours activation defines the overall interaction, but it can be stated that the interaction building block is the behaviours activation between a Source and a Target. In our VE model, three types of virtual entities based on the complexity of the behaviour they exhibit can be defined:

Passive Objects (PO): Virtual entities totally unarmed and passive; their state can be modified only by interaction with the user or with other objects. They are characterized by a behaviour handling their physics, handled by a physic engine, and have no internal logic. Some examples are walls, chairs and cups.

Interactive Entity (IE): It can perceive the surrounding environment and it is associated with an internal logic to decide the type of behaviour to activate accordingly to the perceived input. The types of behaviour are, besides the physic related ones, functional behaviour to reach a goal or send information to other objects. The complexity of the internal logic and the richness of the possible behaviours allow IE to range from very simple to intelligent agents.

User's Avatar: The personification of the user inside the VE. An Avatar can assume different forms, from complete human like 3D shape to simple probe, accordingly to the type of interaction modalities available to the users. For a virtual observer inside the VE that can only perceive the occurring interactions, the Avatar is indistinguishable from another IE and it could be considered as a particular type of IE. Avatar and IEs, if Non Player Character, can be an interaction Source, while any entity can be a Target. In a multimodal interaction more than one behaviour can be activated in parallel, for example a given IE can activate a vibration and a sound to confirm the selection requested from a Source agent.

4 Elements of ASM

ASMs, originally known as Evolving Algebras, were first introduced as a mathematical method to describe discrete dynamic systems aiming to bridge the gap between implementation and specification design models with stepwise transformations of an abstract state by applying rules. The concept of abstract state extends the usual notion of state occurring in Finite State Machines, it is an arbitrarily simple or complex structure, i.e. a collection of domains, with arbitrary functions and relations defined on them. On the other hand, rules are basically nested if-then-else clauses with a set of functions in their bodies. In each state, the conditions guarding rules are checked: all the rules whose conditions are evaluated to true are simultaneously executed, so determining the state transition.

All the parameters are stored in a set of so called locations and at each time the particular configuration of parameters values determines the current state of the ASM. The transition from one state to another is described through a set of formulas:

$$\{\text{if condition}_i \text{ then updates}_i\}_{i=1,\dots,n} \quad (2)$$

Where each condition_i is the guard of the i-th rule, and each updates_i is a set of assignments: $f(t_1, \dots, t_n) = t$. A given function controls the value of a given location based on the actual values of n parameters t_i ; consequently a location can be seen as a pair $(f, (v_1, \dots, v_n))$ where f is the function name and v_i are the values of the t_i parameters. The update is the application of f on the set of v_n defining the new location value v ; it can be resumed with the pair (loc, v) , where loc is the location and v its new value. Different types of functions can be defined, in particular for the purposes of the present work, according to definitions in [17], it is worth considering controlled functions, that are functions updated and only read by the rules of the ASM, monitored functions which are externally updated and only read by the ASM, and out functions, updated but not read by the ASM and available only to the external environment. In order to better manipulate some typical features as the asynchronous parallelism in rules execution or non-deterministic choice, the forAll construct is defined:

$$\text{forAll } x \text{ with } \varphi \text{ rule}(x) \quad (3)$$

meaning that rule(x) is simultaneously executed for every element x satisfying φ . For the unambiguous determination of a next state, it is necessary that updates in each cycle do not conflict each other, this means that a consistency condition must be preserved. We can say that consistency is satisfied in a set of updates “iff” all updates refer to distinct locations. The basic concepts of ASM described above have been extended to support the design of distributed systems; see section 6 for details.

5 Multimodal Interactive Agents

IEs can perceive, at different levels, the world and can act in it. During interaction they behave accordingly to the received input and the current configuration of their internal attributes. From this point of view an IE can be modelled with an agent: it has a set of sensors to perceive the world, a set of effectors to act inside the world, its internal configuration can be described by a set of internal states, it has a decision making component that according to sensors configuration, current state and a set of activation rules, activates certain behaviours through its actuators. We introduce the Multimodal Interactive Agent (MIA) to allow IEs to handle a multimodal interaction. A MIA can be formally defined as follows:

$$\text{MIA} = \langle \text{VO}, \text{SE}, \text{E}, \text{S}, \text{R} \rangle \quad (4)$$

Where, VO is the controlled virtual object as defined in (1). SE is a finite set of sensors, a sensor can be associated with sight, hear and touch, intended as the ability to perceive a contact with another virtual entity. Smell and taste are not covered. The Event sensor can be applied to perceive, under certain constraints, the occurrence of a given asynchronous event. The attribute E is a finite set of effectors. There is a one to one relation between effectors and behaviours; indeed, an effector is the device (physical or virtual) through which the behaviour is actuated. We have four types of behaviours and consequently four types of effectors: Haptic, Acoustic, Dynamic and Visual. The attribute S is the set of possible states in which the agent can pass through. It is defined as all the possible configurations of values of VO attributes. The attribute R defines the Reasoning Manager and its degree of complexity.

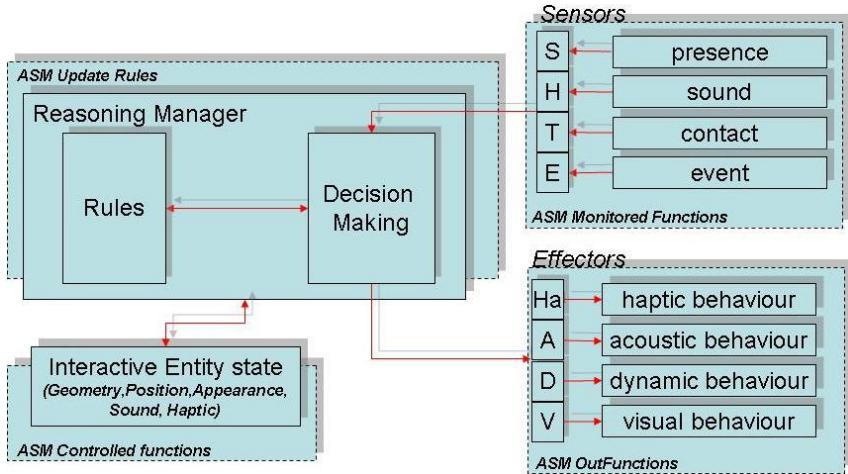


Fig. 1. The MIA architecture: the dotted boxes refer to the respective ASM modeling components

The related architecture is reported in figure 1. The Reasoning Manager is composed of a set of deterministic rules and a Decision Making component. A deterministic rule is an if-then-else construct that, given the current state and the current configuration of sensors, determines which effectors, and consequently which behaviour, must be activated. The Reasoning Manager controls the state of the related IE, in terms of its position, appearance and multimodal displays. Based on the taxonomy reported in [12], the MIA can adopt different behavioural complexity based on the type of controlled virtual entity. For example IEs with deterministic behaviour can be controlled by a simple reflexive based agents in which the Decision Making is absent. A MIA has a high complexity for what concern social ability, indeed it act a direct communication through multimodal interaction.

In modelling a MIA through an ASM the first observation is that each MIA attribute can be stored in an ASM location and handled by an ASM function. In particular, VO attributes are handled by controlled functions, functions describing the locations of the sensors in SE are in (or monitored) functions and functions handling the effectors in E are out functions of the ASM. The set of states of the ASM is equal to the one of the MIA, and the set of update rules can be defined starting from the R attribute in (4) in a quite straightforward way.

One of the strengths of the ASM approach is that a related design methodology has been developed [17] that refine a high level specification called ground model in a top-down procedure. This gives extreme freedom to designer when modelling the Decision Making module, indeed it can be initially resumed with a set of very high level updates and than can be detailed accordingly to the desired agent complexity.

As an example, a deterministic MIA controlling an IE acting as an automatic sliding door will be described. User can open the door with haptic interaction using a point device such as the PHANToM Desktop device from Sensable (see figure 2). The door confirms the selection with a haptic vibration and by changing its colour. The opening procedure can be triggered by pressing a given button, while the haptic



Fig. 2. A multimodal interaction in a VE. The user opens a sliding door with a haptic device. Both the door and the user's haptic probe are controlled with a MIA

probe is in contact with the door. The door must remain in its opening state as long as the user's Avatar is inside a given neighbourhood. The door can trigger its closing procedure only when the user's Avatar has left the neighbourhood. During the closing procedure if the user returns inside this neighbourhood the open procedure must be triggered again. Both for opening and closing procedures the sound of an opening/closing door is played. This MIA can be modelled with an ASM, as follows:

```

Highlight: IF Touched( $\Omega_{sd}$ ) THEN (Vibrate( $\Omega_{sd}$ )  $\wedge$  ChangeColour( $\Omega_{sd}$ ))
Open: IF (OpeningRequest( $\Omega_{sd}$ )  $\wedge$  Closed( $\Omega_{sd}$ )) THEN (MoveDoor( $\Omega_{sd}$ )  $\wedge$  PlaySound( $\Omega_{sd}$ ))
Close: IF(Opened( $\Omega_{sd}$ )  $\wedge$  !InsideBB( $\Omega_{sd}$ )) THEN (MoveDoor( $\Omega_{sd}$ )  $\wedge$  PlaySound( $\Omega_{sd}$ ))
Reopen: IF(Closing( $\Omega_{sd}$ )  $\wedge$  inside( $\Omega_{sd}$ )) THEN (MoveDoor( $\Omega_{sd}$ )  $\wedge$  PlaySound( $\Omega_{sd}$ ))
SlidingDoor: Highlight PAR Open PAR Close PAR Reopen.

```

With rule SlidingDoor being the main rule. For every ASM cycle, every condition is checked in parallel (PAR statement), and the rules associated to true updates are fired simultaneously. In this way only the basic rules must be specified, the others are implicitly defined as a composite parallel execution of these building blocks.

The Avatar' agent is a quite complex process mediating between the behaviour of the human owner and the interaction of the Avatar with the surrounding VE. It is not possible to describe the entire set of Avatar's behaviours with deterministic micro-rules as with the sliding door. At a very abstract level the internal functioning of an Avatar can be resumed with the following actions: perceives the VE, sends the sensed data to the user firing the appropriate user interface behaviours, waits for inputs from the user; if there are inputs then interprets them to infer commands, fires the associated interface behaviours to VE. The relative ASM can be initially described with a very abstract model with the following transition rules, with Avatar being the main rule:

```

Perception: IF configurationNotUpdated( $\Omega_A$ ) THEN PerceiveVE( $\Omega_A$ )
Notification: IF configurationUpdated( $\Omega_A$ ) THEN NotifyUser( $\Omega_A$ )
CheckCommand: IF commandReceived( $\Omega_A$ ) THEN InterpretCommand( $\Omega_A$ )
Execute: IF commandInterpreted( $\Omega_A$ ) THEN ExecuteCommand( $\Omega_A$ )
Avatar: Perception SEQ Notification SEQ CheckCommand SEQ Execute

```

6 An Asynchronous Multi Agent ASM for the VE

In the considered VE model different interactions could take place asynchronously, started by different Sources in the scene; for this reason it is not possible to define a global state intended as a configuration of the system in which no transitions, occurring interactions in this case, are in progress. Despite this, as described in section 3, an interaction is composed by a set of synchronous actions carried out by a Source and a Target, so in any given moment it is possible to univocally describe the state of an interaction. Consequently the global state of the VE can be considered as the sum of single interaction states. Inconsistencies with concurrent interactions can occur when a Target can be interacted by only one Source at a time and when more than one Source can interact with the same Target at a time. In the first case a mutual exclusion mechanism must prevent a second Source to occupy a busy Target; in the second case concurrent interactions must be allowed and monitored to avoid conflicts.

We modelled the overall dynamics of a VE using an async-ASM. This type of ASM is composed by autonomous agents that act independently and are, in turn, internally modelled with basic ASMs. Async-ASM allows consistency checking of an entire dynamic system with the so called coherence condition. We can define an async-ASM run as a set $(M, <)$ of partially ordered moves m ; formally we have:

$$M = \{m_1, m_2, m_3, \dots, m_{k-1}, m_k\};$$

Moves are ordered by the $<$ symbol, meaning that if $m_i < m_j$ then m_i is applied before m_j . Each m_k can be a macro executing a run of a single basic ASM or can trigger multiple basic ASM runs using the construct reported in (3). The coherence condition states that every finite sequential segment X of $(M, <)$ has a state associated $\sigma(X)$. Think of it as the result of all the moves in X with m' executed before m if $m' < m$. $\forall m \in M$ the state $\sigma(X)$ is the result of applying the move m to the state $\sigma(X - \{m\})$ [17]. This means that for every partially ordered sub set of M it must be defined a state that leads to the execution of the most recent move. Let's define $(M', <)$ as:

$$M' = \{m_{s1}, m_{t1}, m_{s2}, m_{t2}\};$$

Where m_{s1} and m_{s2} are execution of rules of a Source agent A and m_{t1} and m_{t2} are execution of rules belonging to a Target agent B. The interaction must always satisfy the coherence condition by allowing the execution of the m_{t2} move. If this condition is satisfied for any given M , corresponding to as much interactions inside the VE, the overall dynamic is synchronized and the async-ASM can continue its run. A mutual exclusion mechanism avoids conflicts: whenever a Source wants to interact with a Target it sends a selection request first, if the Target is already busy the selection request is refused and the Source can wait until the Target returns free or can continue with another interaction. If the selection request is accepted the interaction starts and the related set M' is guaranteed to satisfy the coherence condition. This case stands even for the case in which a given m inside M activates rules belonging to multiple agents by checking each involved Source agent interaction singularly. It can happen that two Sources send a selection requests to the same Target simultaneously, in this case a request will be chosen randomly. Moreover if a Target can be interacted by more than one Source in parallel, a further check is done on the Target type: if it is an

IE the selection stands only for the interaction building block duration then the Target become available to the next Source in a round robin mechanism until each Source has terminated its interaction with the Target. If the Target is a Passive Object, related interactions are physic based that can be handled by the used physic engine: it receives Sources dynamic behaviour data from the async-ASM, such as directions and strength of applied forces in a manipulation tasks, and computes the resulting force to apply to the Target and its new position. The resulting data are sent back to the async-ASM that update the Target location with suited monitored functions. Finally the global async-ASM handling the VE dynamic is as follows:

```
VEASM( $\Omega_{VE}$ ):
While !stop( $\Omega_{VE}$ ) Iterate
    forAll x with  $\varphi_A$  do Agent(x)
    forAll x with  $\varphi_T$  do SelectionChoice(x)
    forAll x with  $\varphi_S$  do
        forAll y with  $\varphi_T$  do
            if SingleSelection(y) then
                MutualExclusionSelection(x)
                Interact(x,y)
            else
                forAll y with  $\varphi_{IT}$  do
                    RoundRobin(y)
                forAll y with  $\varphi_{PT}$  do
                    PhysicInteraction(x,y)
```

Where φ_A is the belonging condition to the MIA agent set, φ_{IT} and φ_{PT} are the belonging conditions to the IE agent and PO sets respectively. It can be noticed how the async-ASM refers only to generic variables and makes no assumptions on their nature. The SingleSelection function checks if the Target can be selected by only one Source at a time. The MutualExclusionSelection and the RoundRobin functions model the respective mechanisms described above, while Interact handles the particular multimodal interaction, finally the PhysicInteraction handles the integration with the physic engine.

7 Conclusion

In this work an approach to describe the dynamic of a VE based on the concept of ASM has been proposed. A VE can be considered as a distributed system in which virtual entities behave and interact in an asynchronous local way. Some of these entities are users' Avatars, others are IE, with which Avatar can interact to receive information, or POs.

Designing and implementing the dynamic of a VEs is quite complex and is usually accomplished with ad hoc solutions such as scripts and graphic languages. The paper proposes MIA agents and ASMs formalisms to model dynamics. Reflexive MIA can be described by basic ASM, while much more effort must be spent to effectively model the user's Avatar due to the nondeterministic behaviour of the human user. For this reason a very high level description has been proposed for the Avatar, letting a more detailed design to successive refinements work. The single ASM allows to describe the behavioural aspect of a given virtual entity while the entire dynamic aspect

is captured with an async-ASM taking single ASMs as internal updatable locations. The async-ASM makes consistency checking by implementing mutual exclusion and round robin procedures.

This work is a first step toward an effective formalization of VEs with ASM in order to give a rigorous engineered method to design behaviours and multimodal interactions in VEs. More effort must be spent to define the functioning of the Avatar and its role inside the model. As various tools exist to simulate ASM execution and make automatic consistency checking and code generation, the next step of our research will implement the design of multimodal VE here described with CoreAsm.

References

- [1] Gurevich, Y.: Evolving Algebras 1993: Lipari guide. In: Börger, E. (ed.) Specification and Validation Methods, pp. 9–36. Oxford University Press, Oxford (1994)
- [2] Bonabeau, E.: Agent-based modeling: methods and techniques for simulating human systems. National Academy of Sciences (1999)
- [3] Salisbury, K., Conti, F., Barbagli, F.: Haptic rendering: introductory concepts. *IEEE Computer Graphics and Applications* 2, 24–32 (2004)
- [4] Pellens, B.: A Conceptual Modelling Approach for Behaviour in Virtual Environments using a Graphical Notation and Generative Design Patterns (unpublished)
- [5] Blackwell, L., von Konsky, B., Robey, M.: Petri net script: a visual language for describing action, behavior and plot. In: 24th Australasian Computer Science Conf (ACSC 2001), vol. 11, pp. 29–37 (2001)
- [6] Willans, J.: Integrating behavioural design into the virtual environment development process. York University, York (2001)
- [7] Fu, D., Houlette, R., Jensen, R.: A visual environment for rapid behavior definition. In: Conference on Behavior Representation in Modeling and Simulation (2003)
- [8] Storrie, H., Hausmann, J.: Towards a formal semantics of UML 2.0 activities. In: Liggesmeyer, P., Pohl, K., Goedicke, M. (eds.) Software Engineering. Lecture Notes in Informatics, vol. P-64, pp. 117–128 (2005)
- [9] Sarstedt, S., Guttmann, W.: An ASM Semantics of Token Flow in UML 2 Activity Diagrams. In: Virbitskaite, I., Voronkov, A. (eds.) PSI 2006. LNCS, vol. 4378, pp. 349–362. Springer, Heidelberg (2007)
- [10] De Boeck, J., Vanacken, D., Raymaekers, C., Coninx, K.: High level Modeling of Multi-modal Interaction using NiMMiT. *Journal of Virtual Reality and Broadcasting* 4(2) (2007)
- [11] Figueiroa, P., Green, M., Hoover, H.J.: InTml: A Description Language for VR Applications. In: Seventh International Conference on 3D Web Technology, Tempe, USA, pp. 15–20 (2002)
- [12] Maher, M., Merrick, K.: Agent Models for Dynamic 3D Virtual Worlds. In: International Conference on Cyberworlds, pp. 27–34 (2005)
- [13] Kallmann, M., Thalmann, D.: Direct 3D Interaction with Smart Objects. In: ACM Symposium on Virtual Reality and Technology, pp. 124–130 (1999)
- [14] Dobrowolski, G.: Programming an Agent as Abstract State Machine. In: Pechoucek, M., Petta, P., Zsolt, L. (eds.) CEEMAS 2005. LNCS (LNAI), vol. 3690, pp. 173–182. Springer, Heidelberg (2005)
- [15] Mi, X., Chen, J.: Agent-based Interaction Model for Collaborative Virtual Environments. In: 9th International Conference on Computer Supported Cooperative Work in Design, pp. 401–404 (2005)

- [16] Mueller, W., Paelke, V.: A Formal Model of a Framework for Simulation Based Animation (unpublished)
- [17] Börger, E., Stärk, R.: Abstract State Machines: A Method for High-Level System Design and Analysis. Springer, Heidelberg (March 2003)
- [18] Reisig, W.: The Expressive Power of Abstract State Machines. Computing and Informatics 20, 1–10 (2003)
- [19] Microsoft FSE Group. The Abstract State Machine Language,
<http://research.microsoft.com/fse/asml/> (last visited March 2006)
- [20] Farahbod, R., Gervasi, V., Glaesser, U.: CoreASM: An extensible ASM execution engine. Fundamenta Informaticae 77, 71–103 (2007)
- [21] <http://oss.sgi.com/projects/inventor/>

A Serialization Algorithm for Mobile Robots Using Mobile Agents with Distributed Ant Colony Clustering

Munehiro Shintani¹, Shawn Lee¹,
Munehiro Takimoto¹, and Yasushi Kambayashi²

¹ Department of Information Sciences, Tokyo University of Science, Japan

² Department of Computer and Information Engineering,
Nippon Institute of Technology, Japan

Abstract. This paper presents effective extensions of our previously proposed algorithm for controlling multiple robots. The robots are connected by communication networks, and the controlling algorithm is based on a specific Ant Colony Clustering (ACC) algorithm. In traditional ACC, imaginary ants convey imaginary objects for classifying them based on some similarities, but in our algorithm, we implemented the ants as actual mobile software agents that control the mobile robots which are corresponding to objects. The ant agent as a software agent guides the mobile robot (object) to which direction it should move. In the previous approach, we implemented not only the ant but also the pheromone as mobile software agents to assemble the mobile robots with as little energy consumption as possible. In our new approach, we take advantage of the pheromone agents not only to assemble the robots but also to serialize them. The serializing property is desirable for particular applications such as gathering carts in airports. We achieve the property by allowing each ant agent to alternatively receive a pheromone agent. We have built a simulator based on our algorithm, and conducted numerical experiments to demonstrate the feasibility of our approach. The experimental results show the effectiveness of our algorithm.

Keywords: Mobile agent, Ant Colony Clustering, Intelligent robot control.

1 Introduction

When we pass through terminals of the airport, we often see carts scattered in the walkway and laborers manually collecting them one by one. It is a laborious task and not a fascinating job. It would be much easier if carts were roughly gathered in any way before the laborers begin to collect them.

In order to achieve such clustering, we have taken advantage of the Ant Colony Clustering (ACC) algorithm which is an Ant Colony Optimization (ACO) specialized for clustering objects. ACO is a swarm intelligence-based method and a multi-agent system that exploits artificial stigmergy for the solution of combinatorial optimization problems. ACC is inspired by the collective behaviors

of ants, and Deneubourg formulated an algorithm that simulates the ant corps gathering and brood sorting behaviors [1]. In ACC, artificial ants collect objects that are scattered in a field, imitating the real ants, so that several clusters are gradually formed.

We previously proposed an ACC approach using mobile software agents. We call it distributed ACC [2,3]. In the approach, some *Ant* agents, which are mobile software agents corresponding to ants, iteratively traverse robots, which correspond to objects picked up by ants. Once the Ant agent migrates to a free robot with no other task, it randomly drives the robot as shown by 1 of Fig. 1. If the robot reaches another robot as shown by 2 of Fig. 1, an Ant agent on the robot locks its robot, and leaves it to look for another free robot. In our approach, the pheromone is also implemented as a collection of mobile software agents. We call them *Pheromone* agents. Each Pheromone agent is created by an Ant agent on a robot included in a cluster. Once it is created on the robot, it duplicates itself and makes the clone migrate to other robots within the scope to disseminate its effect. The Pheromone agent has a vector datum representing strength and direction of its attractiveness, which is used for guiding Ant agents as shown by 3 of Fig. 1. Multiple Pheromone agents reaching the same robot are combined into a single agent with a synthesized vector datum.

Although the previous approach yielded favorable results for its efficiency and energy consumption in the experiments, it just gathered robots, and did not consider how to align them as shown by 4 of Fig. 1. Consider applying the approach to carts in terminals of the airport as mentioned above. After the carts have been roughly gathered, the laborers would have to take them away, for which they would serialize the carts. Such serialization task would be still laborious for the human workers even if the carts are roughly gathered.

We propose a new approach not only gathering robots but also serializing them. In the approach, a pheromone agent on a robot in a cluster initially has a vector value indicating a tail position of the cluster, and when it migrates to another robot,

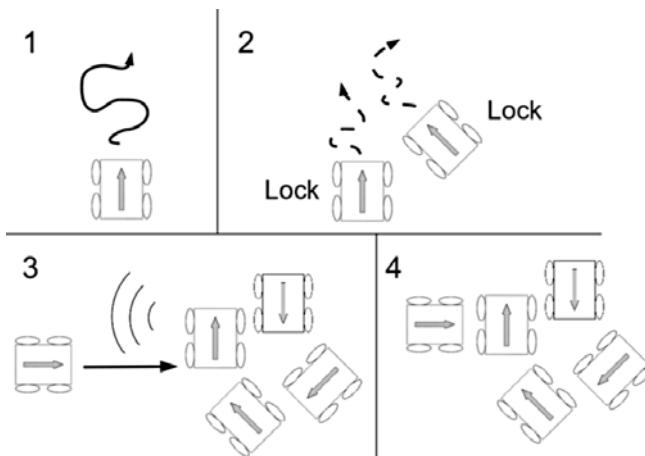


Fig. 1. The outline of the previous algorithm

the pheromone agent on the destination robot has a new vector value which is synthesized from a vector value of the pheromone agent on the source and a vector value indicating the source from the destination. Furthermore, when several pheromone agents migrate to the same robot, an Ant agent on the destination robot picks up one of them instead of synthesizing them. These extensions and modifications for the previous approach enable each cluster generated by distributed ACC to be serialized without sacrificing superior properties.

The structure of the balance of this paper is as follows. In the second section, we describe the background. The third section describes how the new algorithm performs the quasi optimal clustering of the mobile robots and serializing them in each cluster. The fourth section describes the numerical experiments using a simulator based on our algorithm. Finally, we conclude in the fifth section and discuss future research directions.

2 Background

Kambayashi and Takimoto have proposed a framework for controlling intelligent multiple robots using higher-order mobile agents [4][5][6]. The framework helps users to construct intelligent robot control software by migration of mobile agents. Since the migrating agents are higher-order, the control software can be hierarchically assembled while they are running. Dynamically extending control software by the migration of mobile agents enables them to make base control software relatively simple, and to add functionalities one by one as they know the working environment. Thus they do not have to make the intelligent robot smart from the beginning or make the robot learn by itself. They can send intelligence later as new agents. Even though they demonstrate the usefulness of the dynamic extension of the robot control software by using the higher-order mobile agents, such higher-order property is not necessary in our setting. We have employed a simple, non higher-order mobile agent system for our framework. They have implemented a team of cooperative search robots to show the effectiveness of their framework, and demonstrated that their framework contributes to energy saving of multiple robots [6][7]. They have achieved significant saving of energy for search robot applications.

On the other hand, algorithms that are inspired by behaviors of social insects such as ants to communicate to each other by an indirect communication called stigmergy are becoming popular [8][9]. Upon observing real ants' behaviors, Dorigo et al. found that ants exchanged information by laying down a trail of a chemical substance (called pheromone) that is followed by other ants. They adopted this ant strategy, known as ant colony optimization (ACO), to solve various optimization problems such as the traveling salesman problem (TSP) [9]. Deneubourg has originally formulated the biology inspired behavioral algorithm that simulates the ant corps gathering and brood sorting behaviors [10]. Wang and Zhang proposed an ant inspired approach along this line of research that sorts objects with multiple robots [10]. Lumer has improved Deneubourg's model and proposed a new simulation model that is called Ant Colony Clustering [11]. His method could cluster similar objects into a few groups.

3 Serializing Robots

In a new algorithm for serializing robots, an Ant agent and a robot respectively corresponds to an ant and an object of ACC. The Ant agent traverses robots through repeating migrations to find a free robot with no task, as it does in the previous approach [23]. The steps that the Ant agent takes after finding free robot A are as follows. Here AA denotes Ant agent, and PA denotes Pheromone agent:

1. If there is no PA on robot A, an AA makes the robot move randomly as shown by 1 of Fig. 2.
2. If robot A approaches another robot B during random move, A locks itself next to B. This constructs a new cluster as shown by 2 of Fig. 2.
3. If a PA migrates to robot A, the PA guides the AA to the destination represented by its vector datum indicating the back of the robot C on which the PA was originally created as shown by 3 of Fig. 2.
4. Once robot A reaches the destination in a cluster, the AA makes the robot turn to the head of the cluster as shown by 4 of Fig. 2.
5. Robot A is locked there as a member of the cluster as shown by 5 of Fig. 2.

Initial clusters are constructed through the first step and the second step. Once a robot is locked as a member of a cluster, a PA is created on it. In this time, the PA has initial vector value indicating the back of the robot on which the PA resides. When the PA migrates to another robot, its vector value is modified. As shown by Fig. 3, the vector value in which the modification results is calculated

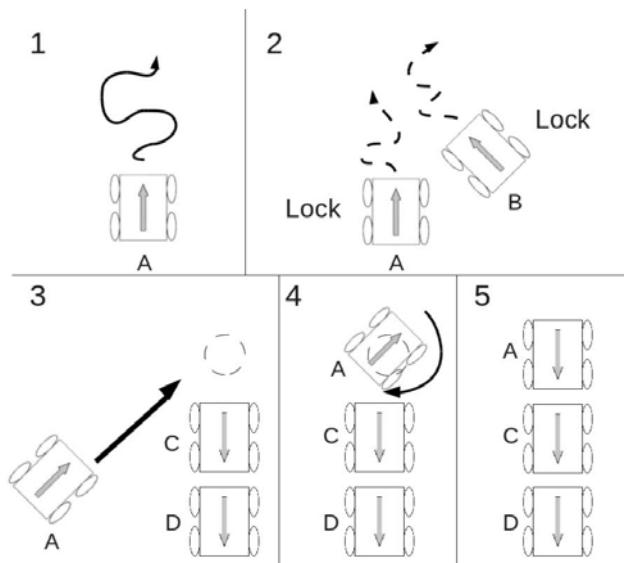


Fig. 2. The outline of the serialization algorithm

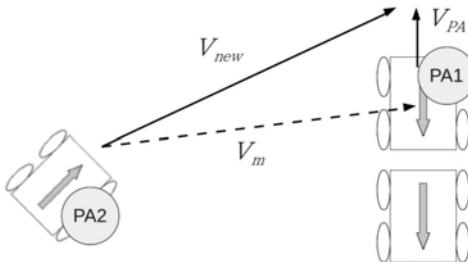


Fig. 3. Calculating the vector value of PA migrating from a locked robot

based on vector value V_{PA} on the source robot of the migration and vector value V_m which indicates the distance and the direction of the source of the migration from the destination, as follows:

$$V_{new} = V_{PA} + V_m \quad (1)$$

It means that the vector value of PA always indicates the back of the robot where the PA was created even if the PA repeated several migrations. Then the PA makes AA drive the robot toward the original robot where the PA was created. The behaviors of the AA with PAs are as follows:

1. If other PAs migrate to the robot with PA, the AA on it picks up the PA that migrated from the nearest robot.
2. Even if the robot approaches another free robot, the PA guides the AA to ignore that free robot and makes the robot reach to the destination.

The first rule makes the behaviors of the robot different from those of in the previous approach when several PAs migrate to the same robot. In the previous approach, the PAs are synthesized by summing their vector values according to the vector calculation. On the other hand, in the new approach, an AA just picks up the PA migrating from the nearest robot. Notice here that the newest PA among PAs born at the same robot is adopted. Each robot needs to be correctly led to the back of another robot in a cluster, while synthesized PA merely led it to any orientation in the cluster. Thus, the rule enables each cluster to be serialized. In detail, the serialization of each cluster is achieved by the following steps. Here robot A denotes the robot approaching to a cluster:

1. An AA on robot A allows a PA on robot B that is the nearest to A (the first robot in the line in Fig. 4) to migrate to A based on the absolute value of its vector datum. The migration results in the vector value indicating the back of B, and therefore, AA is guided for driving A to the destination as shown by 1 of Fig. 4.
2. If there is robot C at the back of B (the second robot in the line in Fig. 4), the AA on A would allow the PA on C to migrate to A. Since A is approaching to the back of B, C becomes the nearest robot from A as shown in 2 of Fig. 4.

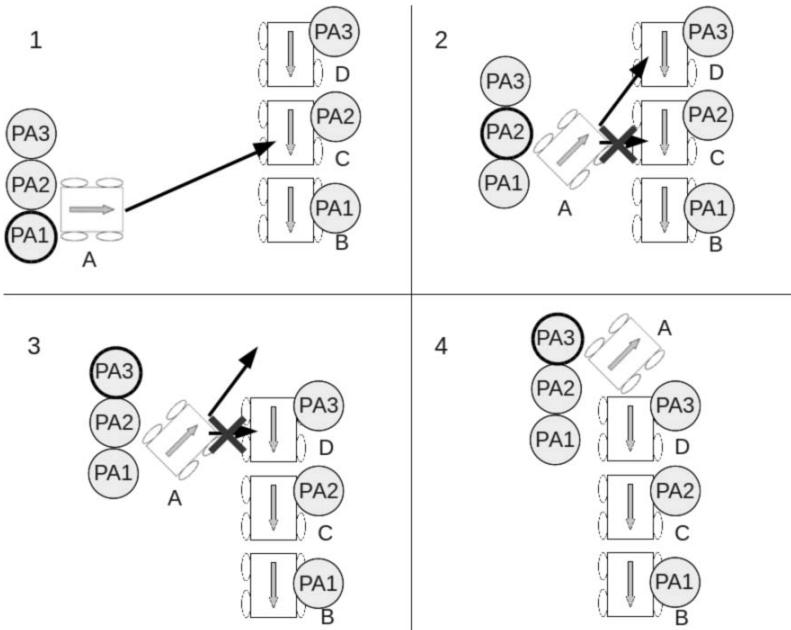


Fig. 4. The process for moving to the tail of a cluster

3. If there is nothing at the back of B, A is locked at the tail position of the cluster as shown in 4 of Fig. 4.

If there is a sequence of several robots behind B, the second step is repeated as the number of robots behind B, and as a result, robot A reaches the tail of the cluster. Finally, robot A at the tail position turns to the head of the cluster as shown by 4 of Fig. 4.

Such a strategy that a robot move to the destination along a lined cluster makes the robot locked wherever the robot approaches a cluster like the previous approach. On the other hand, by the second rule, the AA guided by a PA on the robot makes the robot ignore any clusters except the PA's destination. Thus, the rule contributes to suppressing the negative feature that the first rule gives.

The behavior of AA approaching a cluster is described by the pseudo-code shown in Fig. 5.

4 Experimental Results

In order to demonstrate the effectiveness of our system in a realistic environment, we have implemented a simulator for serializing robots and conducted experiments on it. On the simulator, moving and rotating speed of robots, and lags required in agent migration and object recognition are based on real values in the previous experiments using PIONEER 3-DX with ERSP [237]. In the experiments, we set the following conditions:

```

void doPheromoneWalk(){
    //PA selection
    selectedPA = PAs[0];
    for(i = 1 to n){
        selectedPA = getNearer(selectedPA, PAs[i]);
    }
    //locking
    if(isAroundDestination(selectedPA)){
        sendLockToRobot();
    }
    //AA do Pheromone Walk
    else{
        sendTurnToRobot(selectedPA.vector.direction);
        sendMoveForwardToRobot();
    }
}

```

Fig. 5. The pseudocode of Pheromone Walk

1. Robots are scattered in a 500×500 square field in the simulator.
2. Their initial locations and angles are randomly decided without overlapping.
3. Each robot is represented as a square on the grid field.
4. Locking of each robot is only allowed in the case where it is at the back of the last robot in lined robots.

In the first set of experiments, we have visualized the results of the previous approach and the new one, where two hundreds robots were scattered in the field, and the fifty robots of them had AA. Fig. 6 and 7 show these arrangements respectively. A gray square on the grid denotes one robot, and a circle with the robot at the center shows the scope of the PA on the robot. As shown in the figures, new approach has successfully serialized robots while previous approach has just formed various shaped clusters, although a few branches have occurred because of joining of several lines, and a few bent lines have occurred because of avoiding jutting out the field.

Next, in order to quantitatively discuss these arrangements, we have measured the total length of clusters, where the length of a cluster is the length of a diameter of the minimal circle surrounding the cluster and the average of angles in a cluster. As shown by Fig. 8 and as expected, the total length of the diameters for the new approach is twice as long as previous one. We can observe that the arrangement for the new approach is more line-like than the previous one. The average of angles for previous approach is about 1.5 radian as shown in Fig. 9. That is about $\frac{\pi}{2}$. It means that each robot of a cluster uniformly faces different direction. On the other hand, the average of angles for the new approach is much closer to zero than the previous one. As a result, most robots in a cluster are facing to the same direction.

The second set of experiments, in order to check whether the some good properties of the previous approach are preserved or not, we have conducted several experiments with the different numbers of robots and AAs, and compared their results. As shown in the top of Fig. 10, in the previous approach, the average size of a cluster seem to be about 6 to 8 robots, regardless the number of AAs and

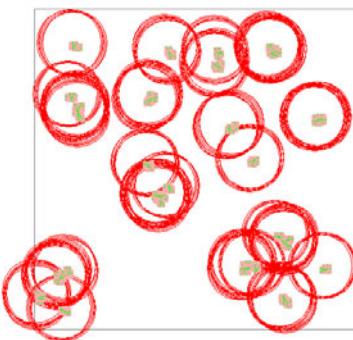


Fig. 6. The result screen of the previous algorithm

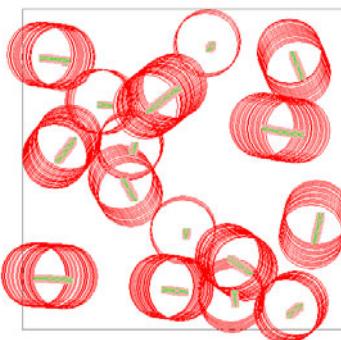


Fig. 7. The result screen of the new algorithm

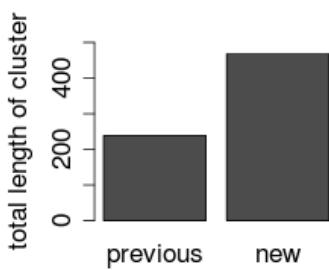


Fig. 8. The total length of clusters

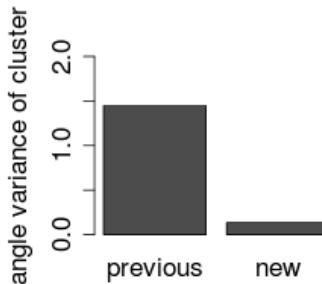


Fig. 9. The angle variance in a cluster

the robots in the field. On the other hand, in the new approach, the average size of a cluster has been gradually increasing with the increase of the number of all the robots. Thus, we can observe that the new approach tends to generate larger clusters. This is because of the property of the new approach that PA makes a robot ignore other clusters except the destination cluster. Considering applying our approach to the arrangement of carts in the airport terminal, it is favorable property that our new approach creates moderately large (long) clusters as the experiments show.

Next, as shown in the center of Fig. 10 in the new approach, we can observe that the time period till convergence for 50 AAs is equal to the time period for 30 AAs though it is less than the time period for 10 AAs, as well as the previous approach. In addition to that, as shown in the bottom of Fig. 10, the less the number of AAs is, the shorter the total length of traces of each robot is. Since the shorter trace means less energy consumption, these results demonstrate that the new approach also has the beneficial features in which the energy consumption can be decreased on some levels without sacrificing efficiency. These results show that the new approach inherits the good properties from the previous approach.

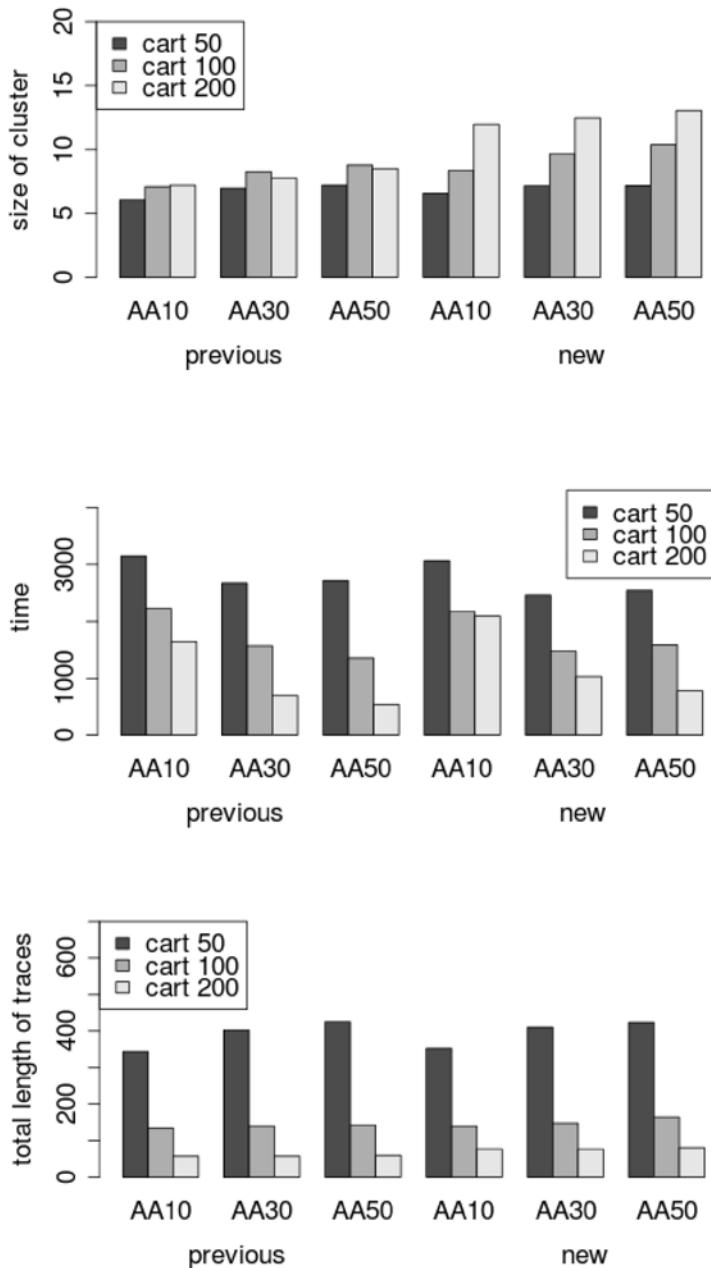


Fig. 10. The size of cluster, the average time taken till convergence and the total length of traces

5 Conclusions

We proposed a serialization algorithm for mobile robots using mobile agents with the distributed ACC. We showed that the algorithm can achieve the serialization of clustered robots, and the algorithm also inherits the good features of the previous algorithm in experiments on a simulation system. On the other hand, we also found that some serialized clusters bent or branched because of avoiding jutting out of the field or joining other clusters.

In order to mitigate these problems, we are designing a new algorithm that makes clusters slowly bent along the edge of the field or other clusters, while preventing the formed clusters being too large. Furthermore, it would be also desirable for each robot not to approach the nearest robot in a cluster but to directly move to the tail of the cluster from the beginning. The new algorithm should integrate this feature too.

References

1. Deneubourg, J., Goss, S., Franks, N.R., Sendova-Franks, A.B., Detrain, C., Chreien, L.: The dynamics of collective sorting: Robot-like ant and ant-like robot. In: Proceedings of the First Conference on Simulation of Adaptive Behavior: From Animals to Animats, pp. 356–363. MIT Press, Cambridge (1991)
2. Mizutani, M., Takimoto, M., Kambayashi, Y.: Ant colony clustering using mobile agents as ants and pheromone. In: Nguyen, N.T., Le, M.T., Świątek, J. (eds.) ACIIDS 2010. LNCS, vol. 5990, pp. 435–444. Springer, Heidelberg (2010)
3. Oikawa, R., Mizutani, M., Takimoto, M., Kambayashi, Y.: Distributed ant colony clustering using mobile agents and its effects. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010. LNCS, vol. 6276, pp. 198–208. Springer, Heidelberg (2010)
4. Kambayashi, Y., Ugajin, M., Sato, O., Tsujimura, Y., Yamachi, H., Takimoto, M., Yamamoto, H.: Integrating ant colony clustering to a multi-robot system using mobile agents. Industrial Engineering and Management Systems 8(3), 181–193 (2009)
5. Kambayashi, Y., Takimoto, M.: Higher-order mobile agents for controlling intelligent robots. International Journal of Intelligent Information Technologies 1(2), 28–42 (2005)
6. Takimoto, M., Mizuno, M., Kurio, M., Kambayashi, Y.: Saving energy consumption of multi-robots using higher-order mobile agents. In: Nguyen, N.T., Grzech, A., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2007. LNCS (LNAI), vol. 4496, pp. 549–558. Springer, Heidelberg (2007)
7. Nagata, T., Takimoto, M., Kambayashi, Y.: Suppressing the total costs of executing tasks using mobile agents. In: Proceedings of the 42nd Hawaii International Conference on System Sciences. IEEE Computer Society CD-ROM, Los Alamitos (2009)
8. Dorigo, M., Birattari, M., Stützle, T.: Ant colony optimization—artificial ants as a computational intelligence technique. IEEE Computational Intelligence Magazine 1(4), 28–39 (2006)

9. Dorigo, M., Gambardella, L.M.: Ant colony system: a cooperative learning approach to the traveling salesman. *IEEE Transaction on Evolutionary Computation* 1(1), 53–66 (1996)
10. Wang, T., Zhang, H.: Collective sorting with multi-robot. In: Proceedings of the First IEEE International Conference on Robotics and Biomimetics, pp. 716–720 (2004)
11. Lumer, E.D., Faiesta, B.: Diversity and adaptation in populations of clustering ants, from animals to animats 3. In: Proceedings of the 3rd International Conference on the Simulation of Adaptive Behavior, pp. 501–508. MIT Press, Cambridge (1994)

A Muscular Activation Controlled Rehabilitation Robot System

Erhan Akdoğan¹ and Zeynep Sişman²

¹ Yıldız Technical University, Mechatronics Engineering Department,
Istanbul, Turkey

eakdogan50@gmail.com

² Bosphorus University, Institute of Biomedical Engineering,
Istanbul, Turkey

zeynep.sisman0@gmail.com

Abstract. The number of people who need rehabilitation increases day by day because of reasons such as laceration, aging, work accidents and etc. Therefore, the need of rehabilitation aids is constantly increasing. There are many research studies about assistive technologies in rehabilitation. Especially, rehabilitation robots have a great importance. Existing rehabilitation robot studies have mostly focused on position and force control. Thus, it is muscular activation that should be evaluated to enhance control results, because the same joint trajectory and/or joint torque can be achieved through different muscular combinations. In this study a muscular activation controlled rehabilitation robot system for lower limbs is proposed. A probabilistic artificial neural network model, which can estimate posteriori probability, was used for discrimination of EMG patterns for robot control with EMG signals.

Keywords: robot, EMG, motion classification, pattern discrimination.

1 Introduction

Spinal cord injury, accidents causing damage in brain or brain vessels and similar diseases cause the need for rehabilitation to grow in the whole world. In parallel to this situation, related technologies are also developing since smart machines are required for supporting physiotherapists in the rehabilitation period. We can classify the systems developed in physical therapy such as smart patient chairs, assisting exoskeletal robots, intelligent orthosis-prostheses (orthotic) and therapeutic exercise robots. The physiotherapy process requires extreme patience from both the patient and the physiotherapist besides being an exhausting and expensive process. Additionally, a physiotherapist can only treat single patient. Nowadays, in order to find solutions to these problems, intelligent therapy equipment is objective in research and development.

In physical therapy and rehabilitation-based health centers, therapy exercise appliances such as CPM [1], BIODEX [2] and CYBEX [3] have been used for a long time. However, these equipments only have a very limited ability to respond to the patient's reaction and to model the physiotherapist's movements. Accordingly, studies on rehabilitation-based robots have increased especially in the last 15 years. Rehabilitation

robots have great importance in the fields of future physical therapy and rehabilitation, due to their features:

- the ability of doing the repetitive movements accurately throughout the therapy
- the ability of measuring the position, the speed and the force and recording these measurements by means of its sensors
- the data of these measurements reflects the result of the therapy.

MIT-MANUS [4], GENTLE/s [5], MULOS [6], ARM-Guide [7], MIME [8] are the most known rehabilitation robots designed for the upper limbs. These designs are developed for the therapy of disorders in motor functions. During the therapy, when the patients cannot do the exercises, the robot arm aids the patient. The effectiveness of robot-aided therapy has been approved by the clinical studies carried out to the present day; refer to [9] and [10]. Besides these studies, rehabilitation robots that aimed at modeling the exercises of the physiotherapist have been developed such as:

- TEM (Therapeutic Exercises Machine) [11]
- REHAROB [12]
- PHYSIOTHERABOT [13]

In all of the studies mentioned above, the force and position sensors detect the patient's reactions. The control algorithms of these robots were developed thorough these position and force feedback data. However, the muscle of the patient is the place where the intention of movement and reactions originally occur in.

When the patient's muscle activations can be evaluated under the control of the rehabilitation robots, the best information about patient's muscular-nervous system will be provided. Thus, this will bring out a more meaningful and more effective control and accordingly a better therapy process. Patient's muscle activation can be obtained by EMG (electromyogram) signals. Various EMG-based systems such as human-machine interfaces, prosthesis, patient chairs, and exterior skeleton robots have been developed to the present; see [14]-[18].

The applications where robot actuators are used as a control signal by means of processed EMG signals are concentrated especially in prosthesis arms and in exoskeletal robots. However, among the rehabilitation robots which aim to recover the motor skills of patients, the use of EMG signals as the robots control signal is limited, especially for lower limbs rehabilitation robots. An assisting robot that is controlled by EMG, BIODEX equipment, was compared with passive exercises in terms of the wrist joint actions. The analysis indicated that a better result is observed with the robot-aided exercises than the passive exercises. The actuators of MIT-MANUS were controlled by means of the EMG signals obtained from the muscles. A game aiming to improve the motor skills was set up on the patients monitor. The patient was told to do the given duties by using his forearm and upper arm together with the robot arm. At the states when the patients muscle signals decreased below a certain level, the robot arm helped the patient with the exercise. In this study four channel EMG signals were used which were obtained from the patients forearm and upper arm regions. In situations when one of these signals exceeded the level of logic 1, robot actuators were activated.

There are only a few studies in the literature about EMG controlled rehabilitation robots for lower limbs. He and Kiguchi proposed an exoskeletal robot manipulator

controlled by EMG signals to assist lower limb motions. This system is wearable and it transfers torques from motors through rigid links to the human joints. They developed a fuzzy-neuro controller to control exoskeletal robot manipulator [19]. Hoshino and et al. developed an assistive device for human locomotion. They used EMG signals to control of gait support mechanism [20].

In this study, we proposed a muscular activation controlled therapeutic exercise robot system for the rehabilitation of the lower limbs. In the system design, basic rehabilitation exercises were references (flexion-extension for knee and hip, abduction-adduction for the hips). A Cybernetic Human-Machine Interface was developed for the control of the proposed system. In this interface, a LLGMN (Log-linearised Gaussian mixture model) algorithm was used for detecting the patient's movements by using EMG signals [21]. The system performs the manual exercises of the physiotherapist besides the passive, active and active assistive exercises. Main differences of proposed system from other rehabilitation robot system for lower limbs are that the proposed system is a therapeutic exercise robot system that is controlled by biofeedback as well as force and position feedback and the developed cybernetic interface can be classify patient motion in order to detect whether correct muscular activation is generated by him.

2 Robot-Aided Rehabilitation System

The structure and the elements of the system are shown in the Fig. 1. According to this, the system is composed of the physiotherapist (PT), the patient, the rehabilitation robot and the cybernetic interface. The PT enters the information about the therapy into the system through the user's interface. This information consist of parameters such as patient's identification (name-surname, date of birth, kind of illness, physical properties of the patient etc.), the limb that will perform the exercise, the kind of the exercise, the therapy period, the number of repetitions of exercise and the therapy's degree of difficulty. Furthermore the EMG electrodes are applied to the appropriate positions on the patient's limb by the PT.

The cybernetic interface is the main control center of the system. It evaluates the patient's EMG signals and the force-position feedback information coming from the rehabilitation robot. Then it produces the torque command which is required for the robot actuators. Regarding this torque command, the rehabilitation robot helps the patient's movement when it is need. More detailed information about the rehabilitation robot and the cybernetic interface are given below.

2.1 Rehabilitation Robot

In this study a robot mechanism with 3 degrees of freedom aimed at the rehabilitation of lower limbs, which was developed by the authors in the earlier studies, was used (figure2). The properties of the robot mechanism are given below. (For deeper information refer to [22])

- It is able to perform active, passive and active assistive exercises as well as model the PT's exercise movements.

- It is a 3-DOF robot manipulator. Thanks to this feature, it can perform the flexion-extension movement for the knee and hip and the abduction-adduction movement for the hip.
- It uses two special force sensors suitable for rehabilitation in order to measure the reacting force of the patient.
- Safety is ensured using both software and hardware.

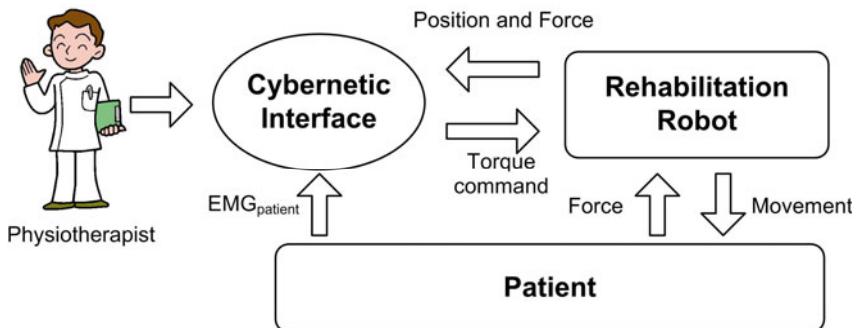


Fig. 1. Robot aided rehabilitation system

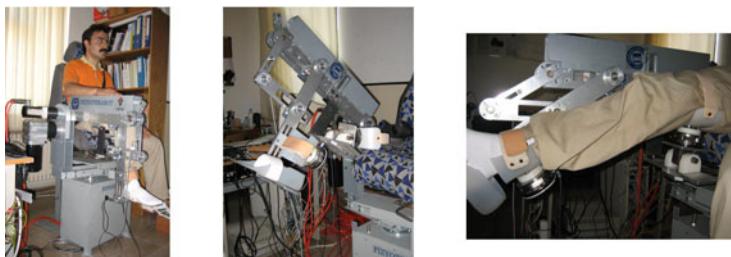


Fig. 2. Rehabilitation robot and positions of force sensors

2.2 Cybernetic Interface

The Cybernetic interface is the control center of the system. The block diagram of the interface is given by figure 3. Explanations regarding its units are given below.

2.2.1 EMG Signal Processing Unit

At the EMG signal processing unit EMG signals coming from the electrodes that are tied to patient's skin are evaluated and used for controlling. This process is performed as follows (see also figure 4). EMG signals coming from L pairs of electrodes are linearized, amplified and filtered respectively. The filtered signals are exemplified sampled with 1 kHz frequency by DAQ cards. Sampled signals are defined as $EMG_i(t)$ ($i = 1, 2, \dots, L$). $EMG_i(t)$ signal is normalized as the sum of the signals coming from L channels electrodes to be 1. The normalized EMG signal is defined in Equation (1).

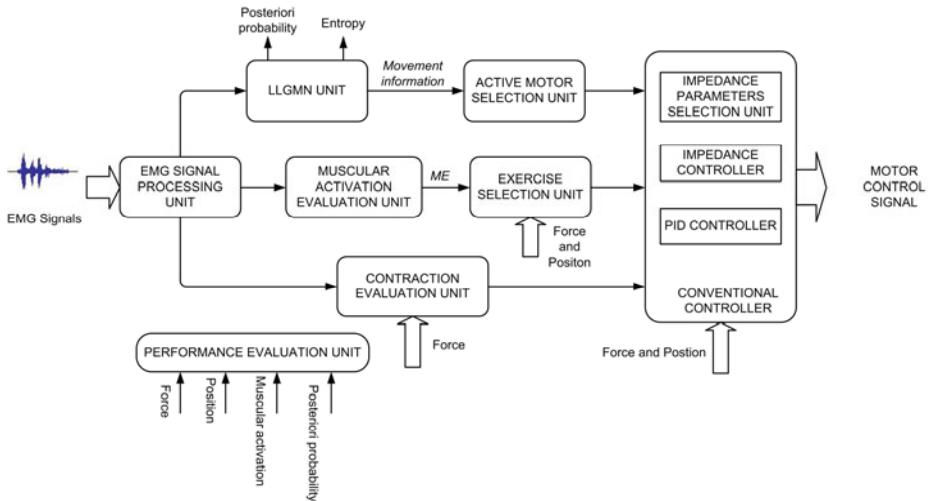


Fig. 3. Cybernetic interface

$$EMG'_i(t) = \frac{EMG_i(t) - EMG_i^{rest}}{\sum_{i=1}^L (EMG_i(t) - EMG_i^{rest})} \quad i = 1, 2, \dots, L \quad (1)$$

In this equation, $EMG'_i(t)$ represents the normalized EMG signals, whereas EMG_i^{rest} represents the mean value of $EMG_i(t)$ of the related limb at rest. Normalized EMG signals are transmitted to related units as shown in Figure 3.

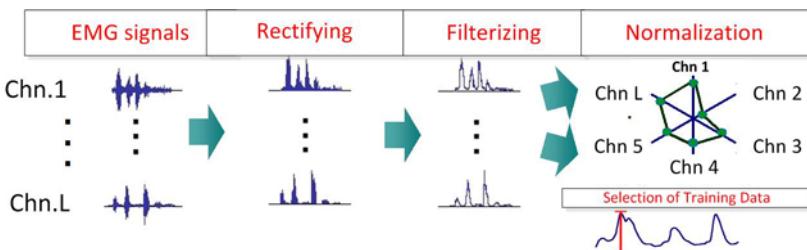


Fig. 4. EMG Signal Processing

2.2.2 Log-Linearized Gaussian Mixture Model Network (LLGMN) Unit

In this study, a probabilistic artificial neural network model, which can estimate posteriori probability, was used for discrimination of EMG patterns in order to determine the EMG-based joint movement [21]. The network structure was built on a statistical model which is composed of log-normal gauss components. Normalized EMG signals are taken by the LLGMN unit for classifying the movements. At first, the signals are processed into a non-linear transformation and then transmitted to LLGMN network model.

The output of the network is the information of the posteriori probability values obtained from the sampled EMG patterns or the information of the movements, the interpreted state of these values. These phases are shown in Figure 5. Besides this, the entropy value is also calculated in order to detect the patterns which cannot be decoupled by the network. If the value of entropy is near to zero, it indicates a right discrimination. If it is near to one, it indicates a wrong discrimination. The network does not give any output in case of exceeding the previously determined entropy value.

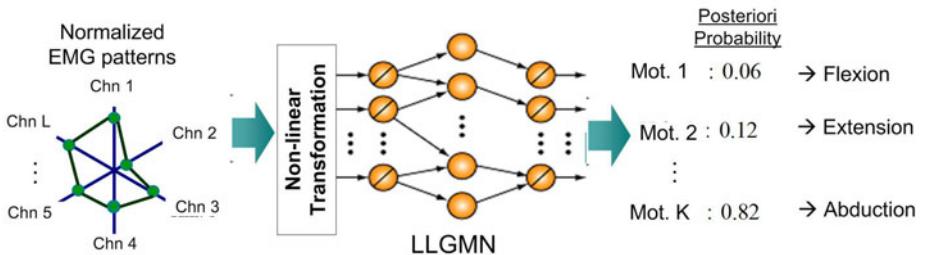


Fig. 5. Log-linearised Gaussian mixture model network (LLGMN)

2.2.3 Muscle Activation Evaluation Unit

The muscle activation (MA) is defined in the equation (2).

$$MA(t) = \frac{1}{L} \sum_{i=1}^L \frac{EMG_i(t) - EMG_i^{rest}}{EMG_i^{max} - EMG_i^{rest}} \quad (2)$$

In this equation, EMG_i^{rest} represents the limb at rest and EMG_i^{max} represents the mean value of $EMG_i(t)$ when the maximum muscle contraction occurs. The calculated value of muscle efficiency is transmitted to the exercise selection unit and is used as a selection parameter for appropriate kind of exercise.

2.2.4 Exercise Selection Unit

The proposed system can perform passive, active-assistive, active and resistive exercises. The exercise selection unit selects the appropriate exercise type through switching method, by using the value of muscle efficiency obtained from the force, position and EMG feedback data and by using the information about the exercise type determined by the physiotherapist.

2.2.5 Contraction Evaluation Unit

A muscle contracture is a shortening of a muscle or joint. It is usually in response to prolonged hypertonic spasticity in a concentrated muscle area, such as is seen in the tightest muscles of people with conditions like spastic cerebral palsy. In case of contracture, this evaluation unit, which is designed for preventing the system from hurting the patient, analyzes contracture. Depending on this analysis, the robots movements are regulated in an appropriate way. Then the necessary information is sent to the conventional control gear.

2.2.6 Active Motor Selection Unit

In this unit, the servo drives which should be allowed are determined in accordance with the movement information coming from LLGMN unit.

2.2.7 Performance Evaluation Unit

This unit is used to evaluate the patient's performance during the rehabilitation session. According to the exercise type, three different performance indices are used. These are EMG pattern index, EMG amplitude index and mechanic parameters index (angle of joint, torque, speed). Each index has two different evaluations: patient index and error index. Patient index reflects only patient's performance, whereas error index detects the error between the patient's performance and the commands sent to patient from the system. In patient index, three different values are calculated: time-dependent instantaneous values, the mean value of time for each trial and the total mean value of all trials in a session.

The normalized EMG signals, the value of muscle efficiency and mechanic parameters (torque, speed, position) are calculated or measured for instantaneous values. By using the time mean values, patient or PT can evaluate the result of the therapy after every trial. The daily therapy evaluation can be done with the total mean value which is calculated at the end of all trials.

The error index indicates the patient's performance ratio and the mean value of the errors in the process.

2.2.8 Conventional Controller

Among the exercises performed by the system, the passive exercises require position control, active-assistive exercises require force and position control, active and resistive exercises require force control. In the proposed system, impedance control method will be used for force control, whereas proportional-integral-derivative (PID) position control method will be used for position control. The conventional control unit will select the control method. Additionally, impedance parameters selection unit, which is located in this unit, will select the appropriate impedance control parameters in accordance with the exercise type.

3 Materials and Method

The system has the two separate working modes learning and therapy. The learning mode aims the patient's movements to be learned by the system by means of EMG signals. With this purpose, EMG electrodes are placed on the related muscles in accordance to (regarding) the limb to be rehabilitated and the kind of movement (flexion-extension, abduction-adduction, etc.). The related movements are done by the patient. For each movement EMG signals are recorded. The learning data is selected from within these recorded signals. The network is trained by the selected data. After the training is completed, the network is ready to classify the movements and the learning phase ends.

In the therapy phase, the patient is attached to the rehabilitation robot without replacing the position of the EMG electrodes. The related exercise type is selected by PT. In therapy stage, the duties of the patient are reflected to the patient's monitor.

The patient is asked to perform these duties, although he is attached to the robot arm. The cybernetic interface, which evaluates the EMG signals and force-position data coming from the rehabilitation robot, produces the motor control signal.

Thus, the rehabilitation robot moves. During the active exercises, patients EMG signal level (measured by muscle efficiency) and decrease in force or contracture case are evaluated by cybernetic interface. The rehabilitation robot helps the patient with the exercise as much as required. In case of contracture, in order to prevent injury, the rehabilitation robot moves with appropriate force and position regarding feedback signal levels or stops the whole movement. Rehabilitation robot has also the ability to perform passive exercises and resistive exercises. Also during these exercises the patient's state is controlled continuously by the feedback information. When needed the system can make some changes in the applied force and position. This process increases the software security besides hardware security elements (limit switches, emergency buttons, mechanical limitations).

4 Conclusion

In this study, we proposed a muscular activation controlled rehabilitation robot system for lower limbs. The system is designed to imply the patient's muscle signals to the force-position feedback control method. A probabilistic artificial neural network model, which can estimate posteriori probability, was used for discrimination of EMG patterns for robot control with EMG signals.

Physiotherapists scale human muscles with six different levels from 0 to 5. Zero level represents the muscles with no contraction, whereas level five represents the strongest muscles. The type of the applied exercises change according to this scale [22]. The system can especially be used for rehabilitation of the patients with muscle levels 0, 1, 2 and 3. There may occur some problems in classifying the EMG signals obtained from the patients with 1 and 2 level muscles, because the patient cannot use his muscles properly and contracture may occur. For this reason, the patients who will use the system will require training which will provide them the knowledge to use their muscles properly. This training will be constituted by games based on performance. In the future study, the performance of the proposed robot-assisting system will be tested by healthy subjects and then patients.

References

1. Salter, R.B., Simmonds, D.F., Malcolm, B.W., Rumble, E.J., MacMichael, D., Clements, N.D.: The biological effect of continuous passive motion on the healing of full thickness defects in articular cartilage: an experimental investigation in the rabbit. *The Journal of Bone and Joint Surgery* 62(8), 1232–1251 (1980)
2. Biodex System 2 User's Manual, Biodex System Medical
3. <http://www.csmisolutions.com> (access time: March 2011)
4. Krebs, H.I., Hogan, N., Aisen, M.L., Volpe, B.T.: Robot-aided neuro-rehabilitation. *IEEE Trans. Rehabil. Eng.* 6(1), 75–87 (1998)
5. Louieiro, R., Amirabdollahian, F., Topping, M., Driessens, B., Harwin, W.: Upper Limb Mediated Stroke Therapy – GENTLE/s Approach. *Autonomus Robots* 15, 35–51 (2003)

6. MULOS Project, <http://www.asel.udel.edu/robotics/newsletter/showcase12.html> (access time: February 11, 2005)
7. Reinkensmeyer, D.J., Kahn, L.E., Averbuch, M., McKenna-Cole, A.N., Schmit, B.D., Rymer, W.Z.: Understanding and treating arm movement impairment after chronic brain injury: Progress with the ARM Guide. *Journal of Rehabilitation Research and Development* 37(6), 653–662 (2003)
8. Lum, P.S., Burgar, C.G., Kenney, D., Van der Loos, H.F.M.: Quantification of force abnormalities during passive and active-assisted upper-limb reaching movements in post-stroke hemiparesis. *IEEE Trans. Biomed. Eng.* 46(6), 652–662 (1999)
9. Ferraro, M., Palazzolo, J.J., Krol, J., Krebs, H.I., Hogan, N., Volpe, B.T.: Robot aided sensorimotor arm training improves outcome in patients with chronic stroke. *Neurology* 61(11), 1604–1607 (2003)
10. Lum, P.S., Burgar, C.G., Kenney, D., Van der Loos, H.F.M.: Robot assisted movement training compared with conventional therapy techniques for the rehabilitation of upper limb motor function following stroke. *Arch. Phys. Med. Rehabil.* 83(7), 952–959 (2002)
11. Sakaki, T., Okada, S., Okajima, Y., Tanaka, N., Kimura, A., Uchida, S.: TEM: Therapeutic exercise machine for hip and knee joints of spastic patients. In: Proceeding of Sixth International Conference on Rehabilitation Robotics, pp. 183–186 (1999)
12. REHAROB Project (2000), <http://rehabrob.manuf.bme.hu>
13. Akdogan, E., Taçgın, E., Adli, M.A.: Knee rehabilitation using an intelligent robotic system. *Journal of Intelligent Manufacturing* 20(2), 195–202 (2009)
14. Zecca, M., Micera, S., Carrozza, M.C., Dario, P.: Control of multifunctional prosthetic hands by processing the electromyographic signal. *Crit. Rev. Biomed. Eng.* 30, 459–485 (2002)
15. Wheeler, K.R., Jorgensen, C.C.: Gestures as input: Neuroelectric joy-sticks and keyboards. *Pervasive Comput.* 2(2), 56–61 (2003)
16. Fukuda, O., Tsuji, T., Kaneko, M., Otsuka, A.: A human-assisting manipulator teleoperated by EMG signals and arm motions. *IEEE Trans. Robot. Autom.* 19(2), 210–222 (2003)
17. Moon, I., Lee, M., Chu, J., Mun, M.: Wearable EMG-based HCI for electric-powered wheelchair users with motor disabilities. In: Proc. IEEE Int. Conf. Robot., pp. 2649–2654 (2005)
18. Arachchilage, R., Gopura, R.C., Kiguchi, K.: EMG-Based Control of an Exoskeleton Robot for Human Forearm and Wrist Motion Assist. In: IEEE International Conference on Robotics and Automation Pasadena, USA (2008)
19. Hel, H., Kiguchi, K.: A Study on EMG-Based Control of Exoskeleton Robots for Human Lower-limb Motion Assist. In: 6th International Special Topic Conference on ITAB, Tokyo, pp. 292–295 (2007)
20. Hoshino, T., Tomono, M., Suzuki, T., Makoto, S., Mabuchi, K.: A Gait Support System for Human Locomotion without Restriction of the Lower Extremities: Preliminary Mechanism and Control Design. In: Proceedings of the 28th IEEE EMBS Annual International Conference, New York City, USA, August 30–September 3, pp. 2950–2953 (2006)
21. Tsuji, T., Fukuda, O., Ichinobe, H., Kaneko, M.: A Log-Linearized Gaussian Mixture Network and Its Application to EEG Pattern Classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part C* 29(1), 60–72 (1999)
22. Akdogan, E., Adli, M.A.: The Design and Control of a Therapeutic Exercise Robot for Lower Limb Rehabilitation: Physiotherobot. *Mechatronics* 21(3), 509–522 (2011)

Design of a Control System for Robot Shopping Carts

Takafumi Kohtsuka¹, Taishi Onozato¹, Hitoshi Tamura²,
Shigetomo Katayama¹, and Yasushi Kambayashi¹

¹ Department of Computer and Information Engineering,
Nippon Institute of Technology, Japan

² Department of Innovative Systems Engineering,
Nippon Institute of Technology, Japan

Abstract. In order to assist elderly or disabled people in supermarkets, we have developed a control system for shopping carts that automatically follow their users. This system liberates elderly and disabled people from the burden of pushing shopping carts, because our proposed shopping cart is essentially a type of autonomous mobile robot that recognizes its user and follows him or her. In this paper, we describe the control system of a novel robot shopping cart that follows the user accurately and automatically. The accuracy is achieved by the infrared laser beam that the cart emits. The cart has a laser range sensor so that it can measure the position and distance of its user. The autonomous mobile robot shopping cart is equipped with an evasion system to prevent collisions with other people, store shelves or other obstacles. The robot shopping cart control system works by adapting itself to a general shopping cart by equipping it with driving part and process computer. Therefore, the device can be detached and can be used in different stores. We developed, implemented, and evaluated the robot shopping cart equipped with this system, and conducted numerical experiments on a simulator. The simulator uses the real data we have collected from the robot shopping cart we have built. The results of the experiments demonstrate that the robot shopping cart is feasible in real store environment.

Keywords: robot control systems, autonomic mobile robot, laser range sensor (LRS), automatic mapping system, self-position estimation.

1 Introduction

We see shopping carts in the supermarket everyday. Customers are pushing around them to carry merchandises. They usually push the carts by using both of their hands. Therefore, if the customer has only one hand, or she has to hold her child's hand, pushing carts is a real burden for her. If she has two or more children, pushing shopping cart is almost impossible. In order to ameliorate the situation and assist the disabled, we have developed an automatic shopping cart that automatically goes after the user of the cart. By using our robot carts, it is possible for customers to pick up merchandises using both hands and to

scrutinize the merchandises. Further, this system enables even people who cannot push a cart to enjoy shopping without human assistants.

The authors have studied robot shopping carts (Fig. 1) [1] [2] [3] [4]. This robot cart goes after its user using Laser Range Sensor (hereafter referred to as LRS). So far our robot shopping cart successfully follows its owner.

Even though this type of robot can successfully track and chase its user, but the ability is not perfect. There are many obstacles in a supermarket, such as shopping shelves and display tables. They interfere in the chase of the robot carts. The presence of other customers also prevents the cart from recognizing its owner. Autonomous mobile robots that successfully work in a super market must satisfy the following conditions. First, they must recognize the wall and the shelf as obstacles. Second, they must identify the moving objects such as people (other customers) as obstacles. In order to achieve these goals, the robot needs to possess a collision avoidance system.

Recently, robotics research is active in various fields due to increasing demands for autonomous mobile robots [5] [6]. We can see them as a part of everyday life such as conveyer robots in indoor plants and security robots in offices [6] [7] [8]. However, robots used in retail stores are rare and unique [9] [10]. In retail stores, location for autonomous robots is extremely difficult in conventional ways. Because there are too many people come and go and shelves are lined up with the same regularity; it is too easy to be lost in a supermarket. In order to develop a practical shopping cart robot, we need to make them adhere to its owner as well as make them avoid any obstacles. So far we are only aware of one robot shopping cart project that is the one conducted at the Tokyo University of Science [9] [10]. Theirs is the most closely related work. Their robot shopping cart employs a web camera in order to track its user. Therefore it suffers the narrow range of view, and consequently it easily loses the sight of the user. Our system demonstrates superior effectiveness to track the human owner. Because LRS has much wider detection area than any web cameras can cover. We describe the LRS we employ in the next section. In this paper we present the development of a shopping cart robot that follows its owner. In addition to the user tracking system, we are aware of the importance of collision avoidance and preventing run-away system as well as durable batteries. The shopping cart robot is designed to stop when it is get lost to prevent undesirable movements. The battery we employ can serve



Fig. 1. The shopping cart robot

the shopping cart about three hours, and the duration time is more than enough for a single shopping activity.

The structure of the balance of this paper is as follows. In the second section, we present the control system that enables the robot cart to follow the user. It intensively uses LRS to locate the user. In the third section, we present the obstacle avoidance system that enables the robot cart to avoid. The fourth section reports the numerical experiments that demonstrate the feasibility of our robot shopping carts. Finally we conclude our discussion and propose future developments in the fifth section.

2 The Control System That Uses LRS to Follow Its User

Our shopping cart robot uses LRS “URG-04LX (made by Hokuyo Electric Machinery Ltd.)” (see Fig. 2) to recognize surrounding circumstances such as the owner that the robot chases and obstacles. This sensor can detect objects in the distance of 4m and in the azimuth of 240 degree on a horizontal plane by using the infrared ray laser light almost in real-time. Fig. 3 shows experiment environment and the obtained image by the sensor.

2.1 Judging the User Position

The following technique is used to judge owner’s position. First, the system analyzes the data acquired from LRS and detects the boundary of user’s both sides, then calculates the central angle between boundaries. This makes the system recognize the user’s position. The robot follows the owner by this method (Fig.4). We have observed that detection is accurate enough in the 4m range, and not affected by obstacles. Fig. 5 shows the result of an experiment that judges the position of the user. The system ignores the other customer located in the left-hand side.

2.2 Calculation of the Motor Operation Amount

The velocity of the motor of the shopping cart robot is determined based on the distance of the user and the following cart. The control system tries to make the cart robot always keep the distance of 1m behind the person whom it follows. The motor itself has a microcomputer and uses it for controlling its operation in PWM. The microcomputer calculates how much speed the motor should rotate

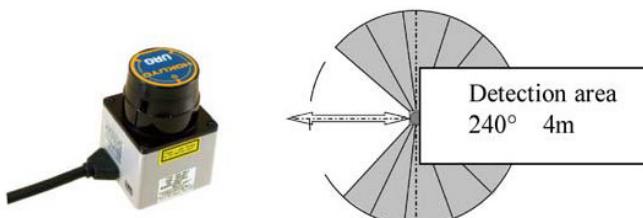


Fig. 2. LRS “URG-04LX”



Fig. 3. Obtained image by the sensor

and send it to the motor. The system uses the proportional control formula (1) to calculate the rotation velocity. A gradual acceleration and deceleration of the motor can be achieved by this method.

$$R(t) = K_p \bullet (V_{des}(t) - V_{act}(t)). \quad (1)$$

$R(t)$ is the calculated motor rotation velocity. $V_{des}(t)$ is the target motor rotation velocity, and $V_{act}(t)$ is the current velocity. K_p is the proportion control gain. The cart robot has two motors to provide locomotion left and right tires independently.

2.3 Discussion of the Following System

We have evaluated our cart robot system through numerical experiments. We have observed that as long as the user walks straight and her walking speed is less than two kilometers per hour, the robot can successfully follow the user. This constraint comes from two machine constraints: one is the maximum speed of the cart is about two kilometers, and the other is the maximum measurement distance of the LRS is about 4 meters. Therefore, as long as the user walks slowly without abrupt turning, the cart robot chases the user steadily. However, walking less than two kilometers per hour is severe constraint, and we have to improve the motors we employ to drive the cart so that the cart robot can follow whoever walks in natural speed.

When we improve the driving force for the cart robot, we have to take account the safeties. It must avoid any collision. Simple following algorithm may causes various collisions especially in the case that the owner of the shopping cart robot turns into an aisle as shown in Fig. 7. Such situation led us to develop a collision avoidance system.

In order to accommodate the curve and the turnabout of the cart robot, the control system gives different rotation velocities to the right and left motors. It is possible to provide different velocities for right and left motors. When the velocity is set to minus, the motor rotates in reverse direction.

The ratio of curve is calculated from the user's position through the right and left divergence. The angle between the user's position and the center line of LRS is calculated by using the cosine theorem. And then the velocities of left and right tires are obtained by using the formula (2). $V_{desL}(t)$ is the rotation velocity of the left tire, and $V_{desR}(t)$ is that of right tire. Here, k is the constant coefficient we have empirically obtained through experiments. Fig. 6 depicts the situation.

$$\begin{aligned}V_{desL}(t) &= V_{des}(t) + k \bullet \theta \\V_{desR}(t) &= V_{des}(t) + k \bullet \theta\end{aligned}\quad (2)$$

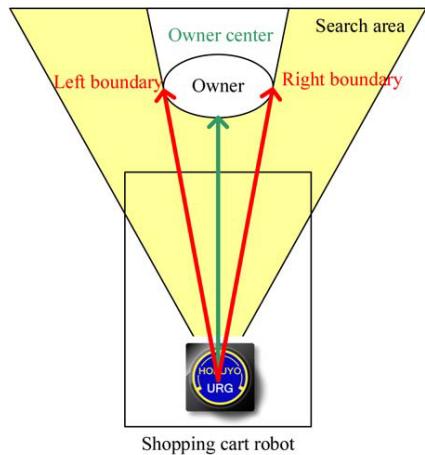


Fig. 4. User detection method by sensor

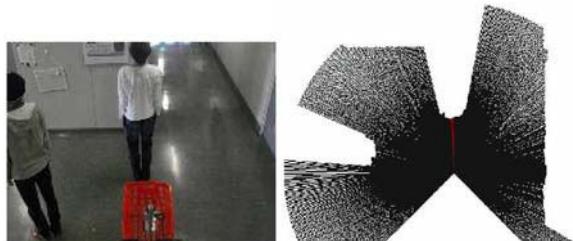


Fig. 5. Judges the position of the user

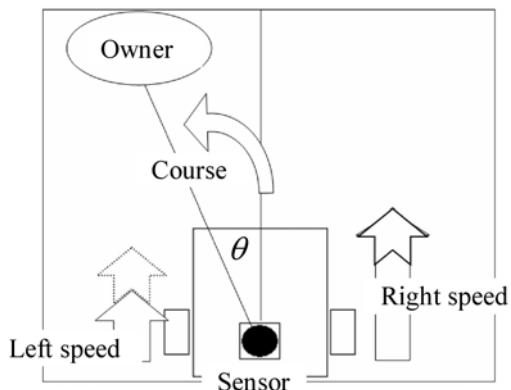


Fig. 6. Speed instruction of motor

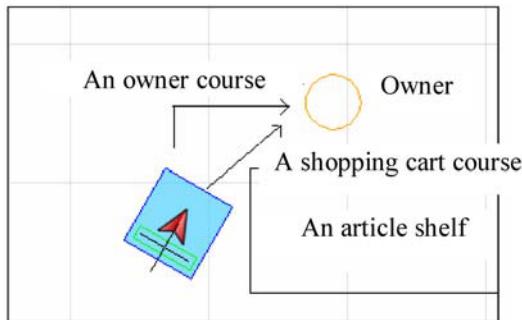


Fig. 7. Colliding case

3 Collision Avoidance System

As the shopping cart robot follows the owner, it finds obstacles such as shelves, or other customers as well as fellow shopping cart robots. The shopping cart robot must avoid the collision with such obstacles in order to achieve its goal. We have integrated a collision avoidance system. We explain that system in the section.

3.1 Obstacle Evasion Technique

Each shopping cart robot has certain collision precaution area in front of it. When the robot finds an obstacle in the collision precaution area, the robot starts evasion movement as shown in Fig.8. The width of the collision precaution

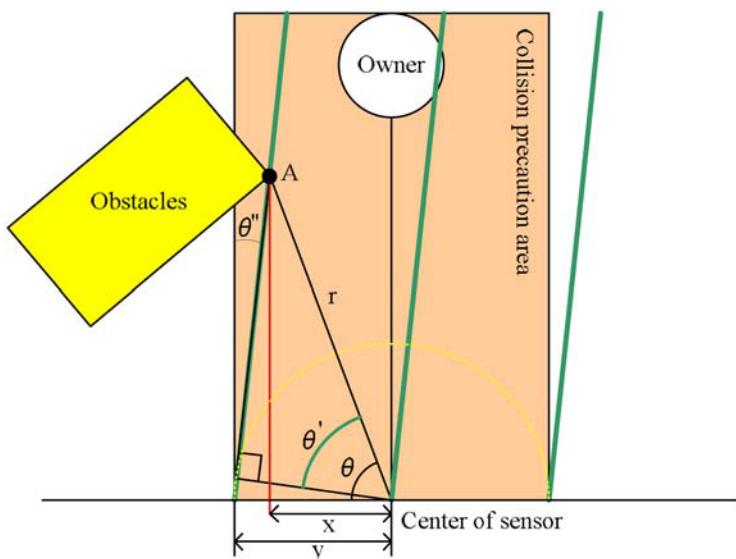


Fig. 8. Calculation of avoided angle

area is a minimum turning circle of the robot, and the height of the area is a distance to the owner. The top point A of the obstacle in Fig. 8 is a point that is nearest from the centerline of the collision precaution area. The A is defined as the point where $|x|$ (shown in formula (3)) becomes minimum value. In order to determine the point A and $|x|$, the calculation of formula (3) is performed for every measurement of r of LRS within the range from 0 to 180 degree within the collision precaution area.

$$|x| = |r \cos \theta| \quad (3)$$

$$\theta'' = \theta - \arccos\left(\frac{y}{r}\right) \quad (4)$$

4 Operational Experiments

In order to demonstrate the feasibility of the robot shopping cart system, we construct a simulator. The simulator employs the real data the real shopping cart robot provides. The numerical experiments using the simulator show our system is feasible in practical setting.

4.1 Operation Experiment

In the experiment, we set the following conditions.

1. The floor has enough friction so that the shopping cart robot does not have to consider any slipping wheels.
2. There is no obstacle whose height is too low for LRS to sense. We assume the LRS can detect all the obstacle the robot encounters.

The user of the simulator can control the customer's movement by using the mouse of the computer. The user is required to move cursor slow and steady enough so that the shopping cart robot can follow.

We have implemented the behaviors of the shopping cart robot in the simulator the same as those of the real shopping cart robot. We have observed that the cart robot successfully follows the owner while avoiding any possible obstacles. We have confirmed that the obstacle evasion system is just effective. The cart robot has problem to enter in narrow aisles. If the passage is narrower than the contact precaution area of the shopping cart, it is impossible for the cart to enter. As shown in Table 1, it is practically infeasible if the passage width is narrower than 120cm. However, then real operational environment, i.e. store floor layout, requires more than 150cm width, the restriction causes no problem.

4.2 Operation Experiment That Uses Store Model

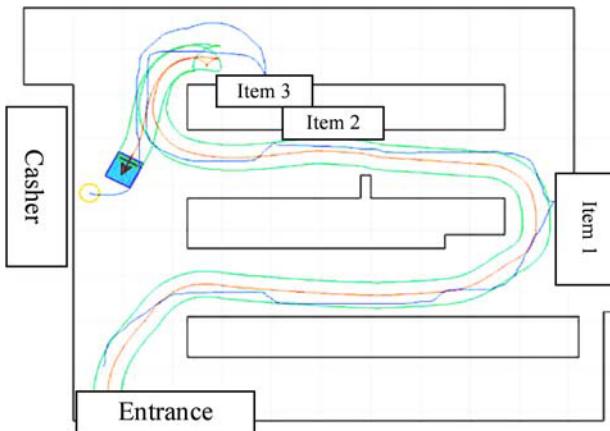
We have conducted a set of experiments in a real general convenience store layout. The scenario is a customer picks the shopping cart robot at the entrance, picks three merchandises at different places, and then come to the cashier to check out as shown in Fig. 9.

Table 1. Results of an experiment in simulator

Width of aisle (cm)	150	140	130	120	110	100
Success rate	100%	96%	94%	40%	8%	2%

As a result of the experiments in the model convenience store, we have observed that the shopping cart robot successfully followed as shown in Fig. 9. The shopping cart robot can not only follow the customer but also turn 180 degree after the customer picks the third merchandise and turns to the cashier. We can conclude that our shopping cart system is practically feasible, and ready to be used in real convenience store.

During the experiments, we tried a few collision detection and avoidance tests. Since the shopping cart robot is designed to keep certain distance, i.e. 1 to 1.5m, from the user to prevent collide with the user from the behind, the same mechanism works when a third customer interfere into them and the cart stops. Sometimes, however, the cart recognizes the interfered person as its owner. We have to solve this problem.

**Fig. 9.** Trace of the shopping cart in a model store

5 Conclusions and Future Works

We have developed a shopping cart robot system. The shopping cart robot successfully follows its owner while it avoids any collision. In order to demonstrate the feasibility of our system, we have constructed a simulator based on data we have collected by the prototyped real shopping cart robots. We are planning to introduce this shopping cart robot into a real convenience store, and collect real data. We are confident that our approach is feasible and contributes to old and disabled people by lessening the burden of carrying merchandises.

One problem we have to solve immediately is the issue of “hijacking” of the robot. When a third person cut across between the owner of the shopping cart and the cart, the robot shopping cart sometimes end-up following the third person. This problem can be solved relatively easily to employ a web camera with simple color recognition capability so that the robot recognizes the color of the clothes of the owner and to ignore any person with different garments. Even with such amelioration, the recognition of the owner cannot be perfect, and the shopping cart may stray from the owner. Within the restriction of shopping cart application, such miss-recognition would not be a big problem. When the owner finds her cart missing, she can always returns and re-catches the cart. In the first place, the ownership of the shopping cart is just temporary.

References

1. Takanashi, Y., Hayakawa, Y., Fukai, W., Tamura, H.: Design and Implementation of an Automatic Tracking Shopping Cart. In: 70th National Convention of IPSJ, vol. 2, pp. 379–380. IPSJ, Tokyo (2008) (in Japanese)
2. Onozato, T., Hishi, S., Tamura, H.: Design and Implementation of Shopping-Cart with Laser Type Measurement Sensor. In: 71st National Convention of IPSJ, vol. 2, pp. 371–372. IPSJ, Tokyo (2009) (in Japanese)
3. Onozato, T., Tamura, H., Katayama, S.: Follow-up Control of Robot for Shopping Cart -Estimation for Searching the Position and Mapping System. In: Ninth Forum on Information Technology 2010 (FIT 2010), vol. 3, pp. 535–538. IPSJ, Tokyo (2010) (in Japanese)
4. Onozato, T., Tamura, H., Kambayashi, Y., Katayama, S.: A Control System for the Robot Shopping Cart. In: IAST International Congress on Computer Applications and Computational Science (CACS 2010), pp. 907–910. IAST, Singapore (2010)
5. Kambayashi, Y., Takimoto, M.: Higher-Order Mobile Agents for Controlling Intelligent Robots. International Journal of Intelligent Information Technologies 1(2), 28–42 (2005)
6. Kambayashi, Y., Ugajin, M., Sato, O., Tsujimura, Y., Yamachi, H., Takimoto, M., Yamamoto, H.: Integrating ant colony clustering to a multi-robot system using mobile agents. Industrial Engineering and Management Systems 8(3), 181–193 (2009)
7. Okusako, S., Sakane, S.: Human Tracking with a Mobile Robot Using a Laser Range-Finder. Journal of Robot Society of Japan 24(5), 605–613 (2006) (in Japanese)
8. Wang, C., Tanahashi, H., Sato, Y., Hirayu, H., Niwa, Y., Yamamoto, K.: Location and Pose Estimation for Active Vision Using Panoramic Edge Histograms. IEICE Journal, D J86-D2(10), 1400–1410 (in Japanese)
9. Nishimura, S., Itou, K., Kikuchi, T., Takemura, H., Mizoguchi, H.: A Study of Robotizing Daily Items for an Autonomous Carrying System. In: Ninth International Conference on Control, Autonomous, Robotics and Vision (ICARCV 2006), pp. 613–618. IEEE Press, New York (2006)
10. Nishimura, S., Mizoguchi, H.: Development of Attachable Modules for Robotizing Daily Items - Person Following Shopping Cart Robot. In: 2007 IEEE International Conference on Robotics and Biometrics, pp. 1506–1511. IEEE Press, New York (2007)

Implementation of the Integrated Management System for Electric Vehicle Charging Stations

Seongjoon Lee, Hongkwan Son, Taehyun Ha, Hyungoo Lee,
Daekyeong Kim, and Junghyo Bae

Smart Power Facility Research Center,
Korea Electrotechnology Research Institute,
70 Bulmosan-gil, Gangnam-gu, Changwon 642-120, Korea

Abstract. We proposed an integrated management system for Electric Vehicle Charging Stations, which were controlled by a remote monitoring server. All the information transferred by the charging stations was stored in the database, and was used for the analysis. Using the accumulated data, the proposed system was able to predict the amount of electricity. The system users could read various reports using a Web browser. In addition, the proposed system could be used in V2G due to its bidirectional communication and remote control abilities.

1 Introduction

With the concerns over fossil fuel depletion and severe environmental problems, many countries were making efforts to produce eco-friendly energy that did not require combustion. The plug-in electric vehicle (EV) allowed the efficient construction of a locally distributed generation system using its mobility and energy storage capacity. Therefore, it was being actively studied worldwide.[\[1,2,3,4\]](#)

To popularize EV, the technical environment must be considered for the establishment of a stable power supply infrastructure. The charging station (CS), which supplied charging service based on a stable power supply, was required for the widespread penetration of EVs. In addition, the realization of the energy transportation feature of EVs required a system that controls the CS so that it could serve as an interface with the power distribution network. Accordingly, an integrated system was required to estimate the required power and manage the CS, efficiently. In this study, a system for CS management was described.

The CS, proposed in [\[1\]](#), didn't have the monitor for the realization of low cost. This function was implemented by using a display device in the vehicle or on the client telephone. However, it was nearly impossible to make the system which can support all kinds of vehicles or telephones. Moreover, for Ubiquitous Environments, it seemed to be more adequate to be connected to the Internet. So, the CS was designed to have the monitor for displaying various events and messages related to the charging, and could have access to the Internet.

The most of legacy system for the charging was intended for a charging stand/station [\[1,3,4,5,6\]](#). However, according to presentation [\[2\]](#), the proper

amount of CSs was 2.5 times the number of EVs. Therefore, a system for monitoring and managing all CSs was necessary. We propose a system that users or managers were able to control CSs through the Internet.

Furthermore, the proposed system collected and stored all data transmitted from CSs. Based on the accumulated data, the manager could predict electric power demand, and can require electric power dispatch in advance. Consequently, the proposed system could supply the stable power source and can be used in V2G due to its bidirectional communication and remote control abilities.

2 Integrated Charging Station Management System

The integrated management system (IMS) for electric vehicle charging stations is used to manage diverse types of discrete CS. Currently, only the quick CS transfers the status information and exchanges charging information via the battery management system (BMS) and the controller area network (CAN). The battery status information can also ensure more efficient charging in a low-voltage environment. In addition, the status exchange with BMS via CAN communication is also being considered for slow CS. To enable the use of the system for V2G, communication with BMS is essential to accurately check the battery capacity and SOC. In the proposed system, a user interface was implemented on a low-power panel personal computer (PC), the OS of which was Microsoft Windows XP to allow CAN communication with BMS and the easy use of CS.

IMS consists of the CS, the aggregator server (AS), the monitoring client (MC), and the PC. AS and MC can be separated or integrated to improve convenience and the systems performance. Fig. 1 shows the system concept that enables integrated management of charging stations.

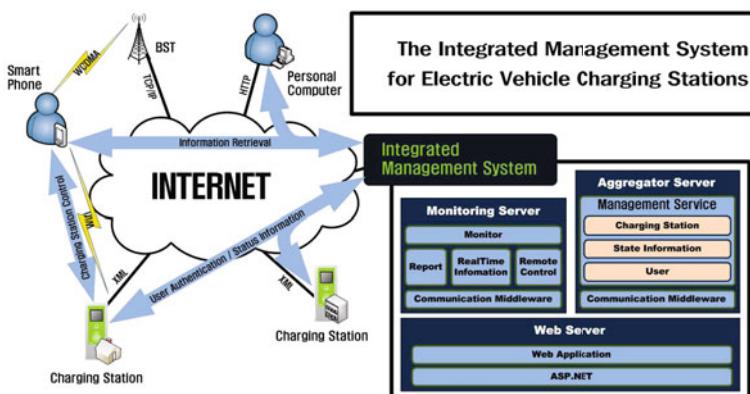


Fig. 1. Overview of the integrated management system for electric vehicle charging stations

2.1 Database Table Design

The tables for efficient data storage are classified into the charging station table (tblChargingStation), the user information table (tblUserInfo), the tag use table (tblTagUse), and the station information table (tblStationInfo).

The tblChargingStation table is used to store the CS installation information. The ID, latitude/ longitude, and a type of CS are stored. After the installation of CS, the administrator receives GPS coordinate of CS and enters it. The IP address, which is enclosed in the data packet that is transferred by CS for the two-way control, is extracted and stored.

tblStationInfo is a table that is used to store CS status data. In it, the high/low temperature, current, and voltage of CS are stored. Using the change in the current, the start and end points of charging can be checked. Through the accumulation of amperage, the required electric energy can be estimated.

tblTagUse is a relationship table that represents the relationship between the user and CS. When the charging is completed, the charging station ID information, tag ID information, total charging energy, and charging time are stored. The tblTagUse table is used to provide the user diverse statistical data.

The tblUserInfo table stores the user information and authentication card information filled up during the user registration process. It includes the address information for the future billing. Fig. 2 shows the relationships between the tables.

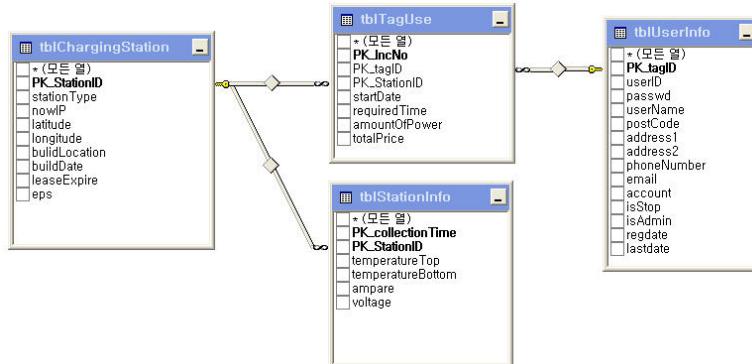


Fig. 2. Relationship between tables

2.2 Charging Station Agent (CS)

CS supplies electric power directly to the electric vehicles. In implementing V2G, however, CS can also deliver the power to the distribution network. For this purpose, two-way control must be possible for CS to provide power bidirectionally. To efficiently manage CS, five message types are defined.

- **Periodic:** This message delivers the charger status information to the AS: i.e., the measured charger temperature, current, and voltage. This information can be used to recognize and arrest the errors and problems of CS.

- **Authentication:** When authentication is required for the user to be able to use CS, this message delivers the user's RFID tag information to AS.
- **StartCharging:** When the authorized user starts charging, the CS delivers the data, including the required energy and charging time, to AS.
- **FinishCharging:** When the charging is completed, the CS delivers the reason for the ending of the charging, the actual charging amount, and the charging time to AS. These messages are accumulated with the periodic messages, and can be used to estimate the electric energy.
- **Response:** When the administrator requests for charger control command or status data transfer using MC, the AS delivers the request message to the corresponding CS. CS processes the request in response, and reports the process results with a response message.

To implement smooth CS communication, the Windows Communication Foundation (WCF) of Microsoft was used for the communication with AS. WCF provides a communication method that can be used in diverse environments without modifying service programs, and enables the user to select the optimal communication method for a given environment.

Fig. 3 shows the operating screen of CS, which is installed in the Korea Electrotechnology Research Institute (KERI) in the republic of korea. To ensure intuitive and easy operation of CS, simple descriptions are displayed on top of the GUI screen. Because CAN communication is not provided for slow charging, the progress is displayed with the requested value as reference. When the battery is fully charged, the current drops to below 0.3 A. Accordingly, the battery is considered fully charged when the current drops to the current value. Therefore, the requested charging amount may differ from the actually supplied amount.



Fig. 3. Operating screenshot of a charging station

2.3 Aggregator Server (AS)

AS is a server that performs a service in response to the request of CS and PC. It provides the following three services.

- **Status information:** The AS analyzes the status information message delivered by CS, and if there is any error or abnormality, it sends an alarm message to MS using Microsoft Message Queue (MSMQ).

- **User:** The user management plays two roles. The first role is to respond to the authentication message delivered from CS. The second role is to provide diverse statistical data when the user accesses AS using the Http protocol.
- **Charging Station:** After the communication status information is stored in each charger, the administrator can request for the current status data on a specific charger from MS, and MS sends the administrator's request to the charger using the demand message.

When events occur, AS delivers all information to MC using MSMQ. Therefore, the administrator can asymmetrically receive messages, and because the messages are stored in the message queue when MC is not in operation, the administrator can check all delivered messages later when MC is operated.

Fig. 4 shows the user authentication process in the simulation environment in KERI. If a user that has an RFID tag that is registered to AS requests for authentication from CS, AS stores the data in the database, and delivers them to the monitoring server. The monitoring server displays the data delivered by AS on the right side of the administrator's GUI screen. The most recent updated information is displayed on top. All the data delivered from CS are transferred and displayed in the same manner as that shown in Fig. 4.

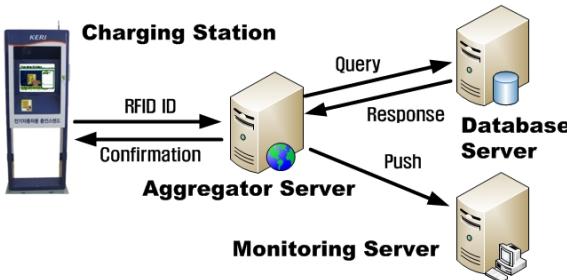


Fig. 4. User authentication process

2.4 Monitoring Client (MC)

MC allows the administrator to recognize the charger status, and to deliver the control commands to the charger. For this purpose, MC has three components.

- **Real-time information:** AS delivers the event information from CS using MSMQ, and MC promptly displays the received information on the screen. Thus, the administrator can recognize all messages delivered from CS in real time.
- **Report development:** The administrator can directly access the database to analyze the stored data, and estimate the required energy or energy flow based on the accumulated data. The users' charging patterns can be analyzed and used for V2G.
- **Remote control:** For the administrator's easy control of CS, MC has the actuator button for charging control on the detailed information window

of CS, and a request button for real measurements during the operation. A detailed actuator and request command for V2G could then be easily implemented later.

Fig. 5 shows the screen that displays the real-time temperature data received from the CS. To search for the detailed status of a specific CS, CS to be controlled must be selected from the tree box on the left. The detailed information window that pops up displays the installation information on the selected CS and the data delivered from CS. In addition, the statistical information tab provides statistics by period.



Fig. 5. Opened window for statistical reports

2.5 Personal Computer (PC)

PC is a personalized terminal that the authorized user can use with a Web browser to inquire on diverse usage statistics, or to manage such statistics. We are developing a system which is monitoring and controlling a CS on a development kit, instead of a smart phone, based on Windows Embedded CE 6.0, as shown in Fig. 6.



Fig. 6. Operating screenshot of a smart device based on Windows Embedded CE 6.0

3 Conclusion

In this paper, a system was implemented to manage and control the charging station bidirectionally, and the system was applied to the charging infrastructure in KERI and in the operation for the efficiency verification and simulation test. The proposed system collected and stored all data transmitted from CSs.

The manager, based on the accumulated data in the IMS database, could predict electric power demand, and can require electric power dispatch in advance. The manufactured management system can be linked with V2G of the smart grid to monitor and control the charging station that enables bidirectional power dispatch.

References

1. Bleijs, C.: Low-cost charging systems with full communication capability. In: 24th International Electric Vehicle Symposium (2009)
2. Cooper, P.: Thinking outside the box: Kaua'i's adaptability in the global energy equation. In: 2008 Kaua'i Renewable Energy Conference (2008)
3. Kil, B., Cho, C., Pyo, Y., Kim, G.: Optimized Strategy of Neighborhood Electric Vehicle with Driving Schedules. Transaction of the Korean Society of Automotive Engineers 18(3), 53–59 (2010)
4. Winkler, T., Komarnicki, P., Mueller, G., Heideck, G., Heuer, M., Styczynski, Z.: Electric vehicle charging stations in magdeburg. In: Vehicle Power and Propulsion Conference, VPPC 2009, pp. 60–65. IEEE, Los Alamitos (2009)
5. Xie, W.D., Luan, W.: Modeling and simulation of public ev charging station with power storage system. In: 2011 International Conference on Electric Information and Control Engineering (ICEICE), pp. 2346–2350 (2011)
6. Zhenpo, W., Peng, L., Tao, X.: Optimizing the quantity of off-broad charger for whole vehicle charging station. In: 2010 International Conference on Optoelectronics and Image Processing (ICOIP), vol. 2, pp. 93–96 (2010)

Input-Output Conditions for Automatic Program Generation Using Petri Nets*

Masahiro Osogami, Teruya Yamanishi, and Katsuji Uosaki

Department of Management Informatioin Science,
Fukui University of Technology,
3-6-1 Gakuen, Fukui-shi, Fukui-pref., 910-8505, Japan
`{osogami,yamanisi,uosaki}@fukui-ut.ac.jp`

Abstract. Recently, demand for software has been increasing due to the influence of factory and office automation, so a software crisis has begun. Concerned with this situation, interest of automatic program generation has been increased. An automatic program generation system MAPP(Module Aided Programming system by Prolog) uses specifications written in easily understood language, and data structures with customized target objects using the generic Prolog module library. During program generation, using input and output conditions of modules, MAPP tries to compensate for any missing specifications when they are not completely described. In this paper, Petri nets will be used corresponds input-output conditions check of MAPP modules. The reachability of Petri net is mathematically analyzable using state equation, algebraic equation or incidence matrices of Petri nets.

Keywords: Input-output conditions, Prolog, automatic program generation, Petri nets, reachability analysis.

1 Introduction

In recent years the study of the improvement of program productivity and maintenance has been actively done. CASE (Computer Aided Software Engineering)[5] has been developed aiming at integration of upper and lower design processes and application of artificial intelligence techniques to automated design process. Various diagrammatic program design techniques have been developed[1][2] and some of them can generate source codes automatically. We have also studied a method of semi-automatic specification refinement and program generation using library modules[3].

From practical point of view, however, it seems that this development is only in its initial stages. It is important to develop automated program design methods which use a language familiar to the user in the object domain, and which have more flexible source code generation techniques.

* This research was supported in part by the Grant-in-Aid for Scientific Research(C) 23560543, Japan Society for the Promotion of Science (JSPS).

It is well known that a new module can be linked to existing modules if its input conditions are in the list of output conditions of the existing modules. In automatic program generation by MAPP, it is necessary to check whether this input-output condition holds or not. For program generation, MAPP tries to compensate for any missing specifications which are not completely described using that input-output conditions of modules. But such input-output conditions checking would be effective only empirically, and be not proved in anyway.

In this paper, we propose a new method using Petri nets for checking input-output conditions for automatic program generation. Since Petri nets can be analyzed mathematically, and then mathematical backgrounds for proving the input-output conditions checking can be obtained.

The remainder of this paper is as follows: In sections 2 to 4, we explain how to generate programs using MAPP (Module Aided Programming system by Prolog) and its construction. In sections 5 and 6, we explain how to check input-output conditions using Petri nets mathematically.

2 Basic Idea of Automatic Program Generation in MAPP

The prototype of the statements and module call functions of C language has a function form:

```
ft function_symbol(ta1 a1, ta2 a2,  
..... , tan an );
```

(1)

where **ft** is a function type, and tai the type of an argument $ai (i = 1, 2, \dots)$. The format is simple and the meaning is clear to the machine if executable machine codes correspond to the call function. However, the function form is hard to understand and to reuse without comments for users. Most importantly, most of module call functions made by users cannot be read and understood without comments. On the other hand, it is easy to read a short comment sentence of a procedure and apply it to a given program design. When a comment sentence corresponds uniquely to a function of C and also to program codes of the body of a module, the call sentence can be written in any format and by any language. Furthermore, the number of basic C functions and the users' generic modules is finite. From the view point of reusing of generic modules, writing and interpretation of comment sentences do not need syntactic and semantic analysis in many cases except for descriptions of test conditions and arithmetic expressions. So, we use simple comment-like sentences as the call sentences of statements and modules. The call sentences are classified by their procedure functions. Based on these set of call sentences, users make up the general specification of the program through the following steps: input of a requesting class name of call sentences; browse of call sentences of the class on a display; search for applicable procedures through the call sentences; specification of an appropriate call sentence by a sentence number; customization of some variables included in the sentence;

and addition of the general specification to a design document. MAPP automatically displays the structure diagram of the design document like SPD(Structured Programming Diagram)[8] to help users' check and plan of the program design. If a call sentence for a requesting procedure can not be found in the dictionary, MAPP can try to synthesize it by combining call sentences of more fundamental functions. Subsequently, it converts the design document to a source code and generates a source program.

3 Construction of Conversion Directories

Statement and module directories are referred to convert call sentences involved in a design document. A directory consists of a predicate, and the argument involves the information necessary for code generation. A statement is a function code provided by a C compiler system and a macro expression, while a module is a user made program unit with several statements and some lower level modules.

3.1 Statement Directories

Expression (2) shows the main part of a statement directory. Here, the input-output condition used for validity checking and for program synthesis is omitted.

```
statement([en(EE,PE),
          arg_type([ARG_TYPE_LIST])]),
```

(2)

while the argument EE and PE in the term en store a call sentence written in English or other informal expressions, and the corresponding predicate expression which generates the C codes, respectively. Here it stores name-type pairs variables in the call sentence.

Expressions(3a) and (3b) show examples of statement directories. A term written in capital letters is a variable term and means that it stands for any possible constant in the defined domain. For example, TYPE in expression (3a) stands for int, float, char, string and others, and means that the C code corresponding to a call sentence is implemented for each type value. The control call sentence of expression (3b) is written in a code like expression in the original directory. We choose the expression as a call sentence, since it can be quickly and easily understood by users.

```
statement(en(['read into',OBJ,'from keyboard by a prompt'],
            prompt_read(OBJ)),
          arg_type([name(OBJ),
                    type(TYPE)])).
```

(3a)


```
statement(en([if(T),then,S1,else,
            S2,end_if],if_els(T,S1,S2))).
```

(3b)

3.2 Module Directories

Each module directory consists of a head part and a tail part as follows

$$\text{module}([\text{HEAD}, \text{TAIL}]). \quad (4)$$

The head part includes a module call sentence, a predicate form and an applicable object type. The tail part includes a module call function written in C, a prototype of the function in the module call sentence, and a name of a file that contains the body part of the function and other components.

Expression (5) shows an example of modules for sort.

```
module([
  en(['sort a ', M, ', * ', N, ' integer type array ', A,
      ' with a key of the ', K, ', th column
      in ascending order by bubble sort'],
      bub_as_sort_ar2i(A,N,M,K)),           ①
  argl([[name(A),type(int),
  array([dim(2),size(N,M)])],
  [name(N),type(int)],...]]),               ②
  [funct(ba_srt2ari(A,N,M,K)),
  prototype([void,ba_srt2ari
  (int[],[],int,int,int)]),                ③
  file_name(sort_ar2),...]).                ④
                                         ⑤           (5)
```

The lines ① and ② belong to the head part, while ③, ④ and ⑤ belong to the tail part. The file name in ⑤ is written in the include statement of a program which uses the module.

4 Automatic Program Generation [4]

Program generation consists of procedural code in modules and declarative statements of data structures.

4.1 Overview of Code Generation

Here we describe a method of code generation from design documents. A design document consisting of call sentences is stored at the argument of a predicate `proc_list` in a list form. MAPP processes the document of every call sentence by applying the following rule recursively:

$$\begin{aligned} \text{proc_list}([H|T]):- \\ & \quad \text{proc}(H), \text{proc_list}(T). \\ & \quad \text{proc_list}([]). \end{aligned} \quad (6)$$

MAPP searches for a conversion directory including the current call sentence. As shown in rule expression (7), if a statement includes the current call sentence X in the term `en(X,Y)` and the object data type coincides with that of the statement directory, MAPP generates the corresponding C expression by `c(Y)` using the predicate form Y of X.

```

proc(X) :-
    statement(S), member(en(X,Y),S),
    member(arg_type([name(OBJ),
                    type(TYPE)],S), obj([name(OBJ),
                    type(TYPE)]),c(Y)).          (7)

```

The objects of control type call sentences consist of several sentences but not terms. Hence, there is no need of type check in conversion of control-type call sentence to the target code expression.

MAPP converts call sentences to target codes in a similar way. Expression (8) shows the main part of a conversion rule of procedure expressions of an array object.

Applying rule (8) to a call sentence X, MAPP searches for module directories which include X in term `en(X,Y)` of the head part. After finding such a module directory, MAPP checks and chooses a module in which the object type coincides with the type of the processed object of the call sentence, where the predicate `p_obj` describes the data structure of the object. Then MAPP adds the C function to the procedure part of the program by ④. Furthermore, MAPP adds a prototype, a real argument name and a file name to the respective temporary storage if they have not been stored yet.

```

proc(X) :-
    module(H,T), member(en(X,Y),H),           ①
    member(arg_type([name(OBJ),
                    type(TYPE),array([dim(D),size(N),
                    max(NMAX)])]),H),
    p_obj([name(OBJ),type(TYPE),
            array([dim(D),size(N),max(NMAX)])]),   ②
    member(funct(FUNCT),T),
    add_funct(FUNCT),                         ③
    .....

```

(8)

4.2 Code Generation of Procedures

When MAPP converts a statement call sentence to a C code, MAPP converts a call sentence X to a predicate form Y given by a term `en(X,Y)` in the statement directory, and then generates a C code by the predicate `c(Y)` flexibly.

The following expressions (9a) and (9b) show conversion rules of statement call expression of `prompt_read` from the predicate form to C code.

```

c(prompt_read(OBJ)) :-
    included('stdio.h'),
    writelist(['printf"',OBJ,'="'],nl,
              c(ead1(OBJ))).          (9a)

```

```
c(read1(OBJ)):-included('stdio.h'),
    p_obj([name(OBJ),type(TYPE)]),
    io_type(TYPE,IO_TYPE),
    writelist(['scanf("",IO_TYPE,"")']),
    write_obj(OBJ),write(''),nl.
```

(9b)

The predicate `included` in (9b) requests that the file name of the argument is added to an include file list if the file name has not been stored yet. The argument `TYPE` of the predicate `p_obj` generated from the processed object designates the corresponding `input_output` type default such as `%d` and `%f` by referring to the `io_type` predicates which provide pairs of `TYPE` and `IO_TYPE`. Conversion rules from a predicate form to C code can be written in a concise form independently of the length of arguments. For examples, the predicate form of a switch call sentence can take any number of cases in the argument as follows:

```
case(INDEX_NAME,[item(1),H1,end_case],
      .....,
      [item(n),Hn,end_case]).
```

(10)

Rules (10a), (10b) and (10c) convert the predicate form (10) to C code.

```
c(case(INDEX_NAME,BODY)):-  
    writelist(['switch(',INDEX_NAME,'){}'],
              nl,case_body(BODY),write'}'),nl.
```

(10a)

```
case_body(BODY)  
    ([[item(NUM),H,end_case]|T]):-  
        writelist(['  case ',NAME,'  ']),  
        proc_list(H),write('  break;'),nl,case_body(T).
```

(10b)

```
case_body([]).
```

(10c)

Call sentence `if_else_if` can be converted in a similar way to generate the function form. To sum up, the conversion rules of the above recursive form convert a call sentence of procedures which include indefinite number of cases and variables to the function form in a top down manner.

4.3 Construction of the Head Part of the Main Program

MAPP constructs the head part of the main program. The head part consists of include statements, a main header, prototype statements of functions and variable declaration. In designing procedures, MAPP stores file names to be included by referring to statement and module directories, and writes out them in C code style.

MAPP also makes declaration of the applied function prototype by referring to their module directories. Variable declarations are made similarly from a variable table in a data structure design document.

Figure 1 shows an example of program that MAPP generates from a procedure design document by referring the data structure.

```

#include <stdio.h>
void main(){
    int year,wy=95,age,cont,i;
    char m;
    printf("cont?=");
    scanf("%d",&cont);
    while(cont!=0){
        printf("m,year?=");
        scanf("%c%d",&m,&year);
        switch(m){
            case 'm': age=wy-(1868+year);
                printf("age=%d\n",age);
                break;
            case 't': age=wy-(1911+year);
                .....
        }
    }
    printf("m,year?=");
    scanf("%c%d",&m,&year);
}

```

Fig.1. An example of programs generated from program design

5 Consideration of Input-Output Conditions Check

Here, we explain the method of conditions check. The modules have their individual input and output conditions. And if a module would be linked, the output condition would be added to the conditions list which keeps all conditions of already linked modules. When a module would be linked, MAPP checks the input condition of the linking module, by investigating whether it has been already included or not in a conditions list of MAPP, which keeps all of the output conditions of already linked modules. Provided the input condition of linking module has been already included in the conditions list of MAPP, the linking module can be linked to target program. And after being linked, the output condition of just linked module would be added to the conditions list of MAPP. But, if the input condition of linking module has not been included in the conditions list of MAPP (meaning the input condition of linking module did not exist in the conditions list of MAPP), the linking would be failed[6][7].

It is important to consider input-output conditions check for automatic program generation. The method mentioned above is only known to be effective based on experience only. So, a new method is proposed here using Petri nets for validation of system conditions of linked modules. The systems described Petri nets are able to be analyzed mathematically.

5.1 Reachability of Petri Nets

Petri nets are a graphical and mathematical modeling tool applicable to many systems[8]. Petri nets are characterized as being concurrent, asynchronous,

distributed, parallel, nondeterministic, and/or stochastic. And we can get the information of systems modeled by Petri nets with analysis of such characteristics. As a graphical tool, Petri nets can be used as a visual-communication aid similar to flow charts, block diagrams, and networks. In addition, tokens are used in these nets to simulate the dynamic and concurrent activities of systems. On the other hand, for a mathematical tool, these properties are analyzable by state equations and algebraic equations. This means it is possible to obtain mathematical backgrounds for the input-output conditions checks.

The $n \times m$ incidence matrix A with rank r of the modeled system can be described as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \uparrow r \quad \uparrow n-r$$

$$\leftrightarrow \quad \leftrightarrow$$

$$m-r \quad r$$
(11)

The matrix B_f can be calculated using A_{11} and A_{12} as

$$B_f = [I_\mu : -A_{11}^T (A_{12}^T)^{-1}] \quad (12)$$

where I_μ is identity matrix with dimension $m - r$.

For the initial marking M_0 and the destination marking M_d , we define

$$\Delta M = M_d - M_0. \quad (13)$$

Then we have the following proposition for the reachability of Petri nets.

[Proposition]

The destination marking M_d can be reachable from the initial marking M_0 , if

$$B_f \Delta M = 0 \quad (14)$$

is satisfied[8].

5.2 An Example of Input-Output Conditions Check Using Petri Nets

Now consider the case of two processes done in sequence. The processes are : read-in process that reads an integer variable from prompt, and print-out process that displays variable.

These read-in and print-out processes correspond to Blocks A and B in a module dictionary in MAPP in Fig.2(a), Processes A and B of automatically generated C program in Fig.2(b), and Parts A and B of the Petri net model in Fig.2(c), respectively. In Fig.2(c), t_1 and t_2 represent the transitions fired by the execution of processes A and B, respectively. Places p_1 and p_3 are correspond to the input condition of t_1 and the output condition of t_2 , respectively. Place p_2 is corresponding to both the output condition of t_1 and the input condition of t_2 .

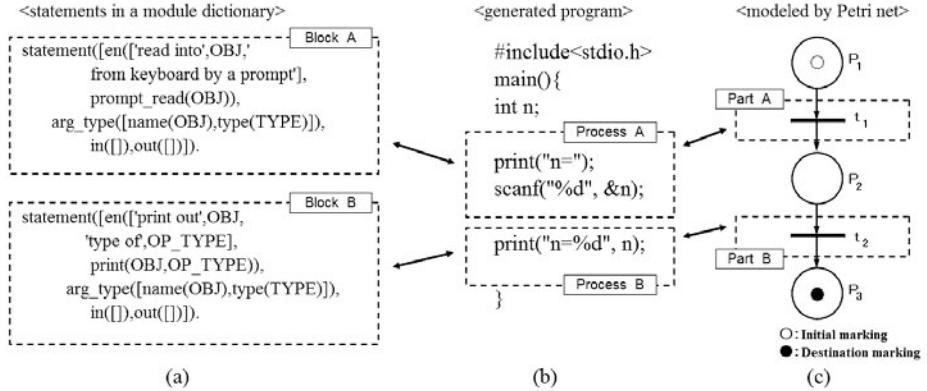


Fig.2. A simple example of a sequential two processes, by general specification, generated program and Petri net

In this case, the incidence matrix of this Petri net is given by

$$A = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad (15)$$

where

$$A_{11} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \quad (16)$$

$$A_{12} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}. \quad (17)$$

Then we have

$$B_f = [1 \quad -1 \quad 1]. \quad (18)$$

For initial marking $M_0 = [1 \ 0 \ 0]^T$ and destination marking $M_d = [0 \ 0 \ 1]^T$, ΔM is given by

$$\Delta M = [-1 \quad 0 \quad 1]^T. \quad (19)$$

Since

$$B_f \Delta M = 0, \quad (20)$$

it is proved that the destination marking M_d is reachable for the initial marking M_0 by the proposition given in section 5.1. This implies that automatically generated program is executable.

Thus the proposed method using Petri nets is shown to be useful for input-output conditions check in automatic program generation.

6 Conclusion

In this paper, we have proposed a new checking method of input-output conditions for automatic program generation. By this method, input-output conditions can be checked mathematically using Petri nets. An example has been presented for illustration.

In further works, state-space-explosion problem should be discussed where large scale complex systems can be dealt with.

References

1. Martin, J., McClure, C.: *Diagramming Techniques for Analysis and Programmers*. Prentice-Hall, Englewood Cliffs (1985)
2. Kai, Y., Endo, Y.: *Structured Programming Diagram Techniques*. Kyoritsu-Pub. (1992) (written in Japanese)
3. Nishida, F., Takamatsu, S., Fujita, Y., Tani, T.: Semi-Automatic Program Construction from Specifications using Library Modules. *IEEE Trans. SE* 17(9), 853–871 (1991)
4. Osogami, M., Nishida, F.: Prolog based Module Retrieval and Program Generation. *Memoirs of Fukui Univ. of Tech.* 23, 313–320 (1993) (written in Japanese)
5. Harada, M.: *The Everything of CASE*. Ohm Corp. (1991) (written in Japanese)
6. Osogami, M., Nishida, F.: A Method of Automatic Program Designing and Soft Code Generation using Informal Procedure Call Sentences. In: Proc. of IASTED - ASC 1998, pp. 161–164 (1998)
7. Osogami, M.: A Method of Automatic Program Generation and Structured Diagram using Informal Procedure Call Sentences. In: Proc. of IASTED - SEA 1999, pp. 104–108 (1999)
8. Murata, T.: Petri Nets:Properties, Analysis and Applications. *Proc. of IEEE* 77(4), 541–580 (1989)
9. Osogami, M.: A study of Input and Output Conditions for Automatic Program Generation. *Memoirs of Fukui Univ. of Tech.* 37, 273–278 (2007) (written in Japanese)

Representation of Knowledge and Uncertainty in Temporal Logic LTL with Since on Frames Z of Integer Numbers

Vladimir V. Rybakov

School of Computing, Mathematics and IT, Manchester Metropolitan University,
Manchester M1 5GD, UK,
and

Institutes of Mathematics, Siberian Federal University, Krasonoyarsk, Russia
V.Rybakov@mmu.ac.uk

Abstract. We study¹ a new hybrid logic $\mathcal{LT}\mathcal{L}_{IA}^{Z,U}$ using as components: LTL based on Z with operations Since and Previous, multi-agent logic with interacting agents, and operation of logical uncertainty. The language of $\mathcal{LT}\mathcal{L}_{IA}^{Z,U}$ contains, together with the standard operations of LTL and multi-agent logic, new knowledge operations **KnI** (for ‘known through interaction’), **GK_L** and **GK_G** (for local and global general knowledge), and expressible logical operations: U - for uncertainty, and U_l for local uncertainty. We consider questions of satisfiability and decidability for $\mathcal{LT}\mathcal{L}_{IA}^{Z,U}$. The key result is construction of an algorithm which recognizes theorems of $\mathcal{LT}\mathcal{L}_{IA}^{Z,U}$ (this implies that $\mathcal{LT}\mathcal{L}_{IA}^{Z,U}$ is decidable, and the satisfiability problem for $\mathcal{LT}\mathcal{L}_{IA}^{Z,U}$ is solvable.)

Keywords: linear temporal logic, multi-agent logic, interaction agents, global and local knowledge, uncertainty, decidability.

1 Introduction, Background

Knowledge Representation (KR) now occupies one of central places in AI and CS; knowledge often is modeled by description logics or by multi-agent logics (cf. e.g. van der Hoek and Wooldridge, [1]). Techniques of KR and multi-agents’ systems are widely used in various areas of AI and IT (cf. for example, Anne Hakansson et al [1], [2]). Multi-agent systems present a conceptual framework for agents, providing a metaphor for our reality visibly populated by such active and more and more intelligent entities. To reason about such systems, we need a formal language. One possibility is to use multi-modal logics (cf. Fagin et al. [9], Halpern and Shore [12]), where modal connectives K_i are used to represent cognitive abilities of individual agents. It is important to develop tools and adequate methods for description and analysis of multi-agent systems. This brings up the question of what basic logic should we use as the foundation of multi-agent reasoning. There is a need to counterbalance expressiveness and simplicity.

¹ This research is supported by Engineering and Physical Sciences Research Council (EPSRC), UK, grant EP/F014406/1.

If a chosen language is too expressive, there is a danger that undecidability can occur (cf. Kacprzak [14] with reduction of decidability to the domino problem). In the simplest case of boolean logic and autonomous agents, decidability usually easily follows, at least for standard systems (cf. [9]).

Temporal logics are currently the most widely used specification formalisms for reactive systems. They were first suggested to specify properties of programs in the late 1970's (cf. Pnueli [16]). The most used temporal framework is the linear-time propositional temporal logic LTL, which has been extensively studied from the point of view of various prospects of applications (cf. e.g. Manna and Pnueli [15], Clark E. et al. [6]). Temporal logics have numerous applications to safety, liveness and fairness, to various problems arising in computing (cf. Barringer, Fisher, Gabbay and Gough [3]). Model checking for LTL formed a direction in logic in computer science, which uses, in particular, applications of automata theory (cf. Vardi [7,33]). The mathematical theory of temporal logics and their semantic theory based on Kripke/Hintikka-like models and temporal Boolean algebras formed a highly technical branch in non-classical logics (cf. van Benthem [32], Gabbay and Hodkinson [10], Hodkinson [13]). Axiomatizations of various (uni)-temporal linear logics are summarized in de Jongh et al. [8].

In this paper we study a new hybrid logic $\mathcal{LTL}_{\mathcal{IA}}^{\mathcal{Z}, \mathcal{U}}$ using as components: LTL based on \mathcal{Z} with operations Since and Previous, multi-agent logic with interacting agents, and operation of logical uncertainty. In particular, we use the knowledge operation **KnI** (for ‘known through interaction’), which is the dual counterpart of the common knowledge operation (cf. Fagin et al. [9]). Our main result is found algorithm for recognizing the theorems of $\mathcal{LTL}_{\mathcal{IA}}^{\mathcal{Z}, \mathcal{U}}$, so in particular, we show that $\mathcal{LTL}_{\mathcal{IA}}^{\mathcal{Z}, \mathcal{U}}$ is decidable. Notice that the result goes through in spite of $\mathcal{LTL}_{\mathcal{IA}}^{\mathcal{Z}, \mathcal{U}}$ itself does not have standard finite model property.

This our paper is based on technique developed in [27] and extends results obtained in [28] and [31] by introduction in the language operations of logical uncertainty, knowledge by interaction and operation since responsible for past. In this paper we essentially use the technique worked out in [27,28,29,30,26,31].

2 Preliminaries: Language and Semantics of $\mathcal{LTL}_{\mathcal{IA}}^{\mathcal{Z}, \mathcal{U}}$

To describe logical language and models, in the beginning, we extend the standard language of *LTL* for modeling agents' reasonings through possible interactions between agents; also we enrich the language with more refined operations, such as strong until and weak until. The basic semantic objects upon which we ground our logic are the following Kripke/Hintikka models. A frame

$$\mathcal{Z}_C := \langle \bigcup_{i \in Z} C(i), R, R_1, \dots, R_m, Next, Prev \rangle$$

is a tuple, where Z is the set of all integer numbers, $C(i)$ are nonempty sets (we assume $C(i) \cup C(j) = \emptyset$ if $i \neq j$).

Relation R is a binary linear relation for time, R_1, \dots, R_m are binary accessibility relations imitating possible agent transitions.

$$\forall a, b \in \bigcup_{i \in Z} C(i) (a R b) \Leftrightarrow [a \in C(i) \& b \in C(j) \& i < j] \vee \exists i \in Z [a, b \in C(i)].$$

Any R_j is a reflexive, transitive and symmetric relation, and $\forall a, b \in \bigcup_{i \in Z} C(i)$, $a R_j b \Rightarrow \exists i \in Z [a, b \in C(i)]$; $a \text{ Next } b \Leftrightarrow [\exists i ((a \in C(i)) \& (b \in C(i+1)))]$. The informal meaning of \mathcal{Z}_C is as follows: each $i \in Z$ (any integer number i) is the time index for the cluster of *all possible states* arising at the step i in the current computation. Each $C(i)$ is a finite set of all possible states in time point i , and R models discrete current of time.

The relations R_j are intended to model accessibility relations of agents (in the clusters of states $C(i)$) at any current time point i . So, as usual, any R_j is supposed to be an *S5*-like relation, i.e. an equivalence relation on clusters $C(i)$. Towards computational aspects, we assume the reasoning/computation is simultaneous and parallel—after a step, a new cluster of possible states appears, and agents will be given new access rules to the information in this time cluster of states. However, the agents cannot predict, which access rules they will have (that is why we do not use *nominals*).

The language we propose uses the standard language of LTL (which extends the language of Boolean logic by operations **N** (next), **U** (until)) and the new LTL-like operations together with multi-agent logic operations extended to handle agent interaction. In total, the operations are **U**, **N**, new binary logical operations **U_w** (weak until), **U_s** (strong until), **S** (since), **S_w** (weak since), **S_s** (strong since), **N⁻¹** (previous), unary knowledge operations **K_j**, $1 \leq j \leq m$, additional unary operations **GK_L**, **GK_G** for local and global knowledge, and the unary operation **KnI** for *to be known by interaction*. The formation rules for formulas are as usual. The intended meanings of the operations are:

K_jφ means the agent j *knows* φ in the current state of a time cluster;

GK_Lφ means that φ is *local general knowledge* in the current state;

GK_Gφ means φ is *global general knowledge* in the current state;

KnIφ: in the current state φ *may be known by interaction between agents*;

Nφ has the meaning φ holds in the *next time cluster* of states (state);

N⁻¹φ means φ holds in the *previous time cluster* of states (state);

φUψ can be read: φ holds until ψ will hold;

φSψ φ says that since ψ was true, φ holds until now;

φU_wψ has the meaning φ *weakly holds* until ψ will hold;

φU_sψ has the meaning φ *strongly holds* until ψ will hold;

φS_wψ φ says that since ψ was true, φ weakly holds until now;

φS_sψ φ means that since ψ was true, φ strongly holds until now.

For any collection of propositional letters *Prop* and any frame \mathcal{Z}_C , a valuation in \mathcal{Z}_C is a mapping which assigns truth values to elements of *Prop* in \mathcal{Z}_C . Thus, for any $p \in \text{Prop}$, $V(p) \subseteq \mathcal{Z}_C$. We will call $\langle \mathcal{Z}_C, V \rangle$ a model (a Kripke/Hintikka model). For any such model \mathcal{M} , the truth values are extended from propositions of *Prop* to arbitrary formulas as follows (for $a \in \mathcal{N}_C$, we denote $(\mathcal{M}, a) \Vdash_V \varphi$ to say that the formula φ is true at a in \mathcal{M}_C w.r.t. V). The rules for specifying truth values are as follows: $\forall p \in \text{Prop}, (\mathcal{M}, a) \Vdash_V p \Leftrightarrow a \in V(p)$;

$$(\mathcal{M}, a) \Vdash_V \varphi \wedge \psi \Leftrightarrow (\mathcal{M}, a) \Vdash_V \varphi \wedge (\mathcal{M}, a) \Vdash_V \psi; \quad (\mathcal{M}, a) \Vdash_V \neg \varphi \Leftrightarrow$$

$$\text{not}[(\mathcal{M}, a) \Vdash_V \varphi]; \quad (\mathcal{M}, a) \Vdash_V \mathbf{K}_j \varphi \Leftrightarrow \forall b[(a R_j b) \Rightarrow (\mathcal{M}, b) \Vdash_V \varphi].$$

As usual, $\mathbf{K}_j \varphi$ says that φ holds in all states available for the agent j .

$$(\mathcal{M}, a) \Vdash_V \mathbf{GK}_L \varphi \Leftrightarrow \forall j \forall b[(a R_j b) \Rightarrow (\mathcal{M}, b) \Vdash_V \varphi].$$

Thus, φ is *local general knowledge* if it holds in all states which are accessible in the *current time point* for every agent. GK_L is more commonly referred to as the *E*-operation, for ‘everyone knows’.

$$(\mathcal{M}, a) \Vdash_V \mathbf{GK}_G \varphi \Leftrightarrow \forall b[(a R b) \Rightarrow (\mathcal{M}, b) \Vdash_V \varphi].$$

Thus, φ is *global general knowledge* if it holds in all states in all future (and current) time clusters.

$$(\mathcal{M}, a) \Vdash_V \mathbf{KnI} \varphi \Leftrightarrow \exists a_{i1}, a_{i2}, \dots, a_{ik} \in \mathcal{M}$$

$$a R_{i1} a_{i1} R_{i2} a_{i2} \dots R_{ik} a_{ik} \& (\mathcal{M}, a_{ik}) \Vdash_V \varphi.$$

Thus, if $\mathbf{KnI} \varphi$ holds, φ is known by interaction between the agents, i.e. there is a path of transitions via the agents’ accessibility relations which leads to a state where φ holds.

$$(\mathcal{M}, a) \Vdash_V \mathbf{N} \varphi \Leftrightarrow \forall b[(a \text{ Next } b) \Rightarrow (\mathcal{M}, b) \Vdash_V \varphi];$$

$$(\mathcal{M}, a) \Vdash_V \mathbf{N}^{-1} \varphi \Leftrightarrow \forall b[(a \text{ Prev } b) \Rightarrow (\mathcal{M}, b) \Vdash_V \varphi];$$

$$(\mathcal{M}, a) \Vdash_V \varphi \mathbf{U} \psi \Leftrightarrow \exists b[(a R b) \wedge ((\mathcal{M}, b) \Vdash_V \psi) \wedge$$

$$\forall c[(a R c R b) \& \neg(b R c) \Rightarrow (\mathcal{M}, c) \Vdash_V \varphi)];$$

$$(\mathcal{M}, a) \Vdash_V \varphi \mathbf{U}_w \psi \Leftrightarrow \exists b[(a R b) \wedge ((\mathcal{M}, b) \Vdash_V \psi) \wedge$$

$$\forall c[(a R c R b) \& \neg(b R c) \& (c \in C(i)) \Rightarrow \exists d \in C(i)(\mathcal{M}, d) \Vdash_V \varphi)];$$

$$(\mathcal{M}, a) \Vdash_V \varphi \mathbf{U}_s \psi \Leftrightarrow \exists b[(a R b) \wedge b \in C(i) \wedge$$

$$\forall c \in C(i)((\mathcal{M}, c) \Vdash_V \psi) \wedge \forall c[(a R c R b) \& \neg(b R c) \Rightarrow (\mathcal{M}, c) \Vdash_V \varphi)];$$

$$(\mathcal{M}, a) \Vdash_V \varphi \mathbf{S} \psi \Leftrightarrow \exists b[(bRa) \wedge ((\mathcal{M}, b) \Vdash_V \psi) \wedge$$

$$\forall c[(bRcRa) \& \neg(cRb) \Rightarrow (\mathcal{M}, c) \Vdash_V \varphi]];$$

$$(\mathcal{M}, a) \Vdash_V \mathbf{S}_w \psi \Leftrightarrow \exists b[(bRa) \wedge ((\mathcal{M}, b) \Vdash_V \psi) \wedge$$

$$\forall c[(bRcRa) \& \neg(cRb) \& (c \in C(i)) \Rightarrow \exists d \in C(i)(\mathcal{M}, d) \Vdash_V \varphi]];$$

$$(\mathcal{M}, a) \Vdash_V \mathbf{S}_s \psi \Leftrightarrow \exists b[(aRb) \wedge b \in C(i) \wedge$$

$$\forall c \in C(i)((\mathcal{M}, c) \Vdash_V \psi) \wedge \forall c[(bRcRa) \& \neg(cRb) \Rightarrow (\mathcal{M}, c) \Vdash_V \varphi]].$$

To incorporate uncertainty, we suggest a new unary logical operator U , and extend formation rules for formulas by: if A is a formula UA is a formula as well. This operation may be nested, so, say $U\neg(A \rightarrow \neg UA)$ is a formula. The meaning of this operation in context of our approach is as follows. We suggest, for any formula φ ,

$$UA := \mathbf{KnI}\varphi \wedge \mathbf{KnI}\neg\varphi.$$

So, φ is uncertain if φ may be visible to be true via agents' interaction and may be visible to be false. Besides, we can use *local* uncertainty operation U_l , defining it as

$$U_l\varphi := [\bigvee_{1 \leq i \leq m} \neg K_i \varphi] \wedge [\bigvee_{1 \leq i \leq m} \neg K_i \neg \varphi].$$

The mining for local uncertainty would be φ is locally uncertain if some agents know that φ is true, and some know that φ is false. Because all suggested operations for uncertainty, as we see, are explicitly defined within standard language we do not need to extend it by new external logical operations.

Given a Kripke structure $\mathcal{M} := \langle \mathcal{Z}_C, V \rangle$ and a formula φ , (i) φ is *satisfiable* in \mathcal{M} (denotation – $\mathcal{M} \Vdash_{Sat} \varphi$) if there is a state b of \mathcal{M} ($b \in \mathcal{Z}_C$) where φ is true: $(\mathcal{M}, b) \Vdash_V \varphi$. (ii) φ is *valid* in \mathcal{M} (denotation – $\mathcal{M} \Vdash \varphi$) if, for any b of \mathcal{M} ($b \in \mathcal{Z}_C$), the formula φ is true at b $((\mathcal{M}, b) \Vdash_V \varphi)$.

For a frame \mathcal{Z}_C and a formula φ , φ is satisfiable in \mathcal{Z}_C (denotation $\mathcal{Z}_C \Vdash_{Sat} \varphi$) if there is a valuation V in the frame \mathcal{Z}_C such that $\langle \mathcal{Z}_C, V \rangle \Vdash_{Sat} \varphi$. φ is valid in \mathcal{Z}_C (notation $\mathcal{Z}_C \Vdash \varphi$) if *not*($\mathcal{Z}_C \Vdash_{Sat} \neg \varphi$).

Definition 1. The logic $\mathcal{LTL}_{IA}^{\mathcal{Z}, U}$ is the set of all formulas which are valid in all frames \mathcal{Z}_C .

We say a formula φ is *satisfiable* iff there is a valuation V in a Kripke frame \mathcal{Z}_C which makes φ satisfiable: $\langle \mathcal{Z}_C, V \rangle \Vdash_{Sat} \varphi$. Clearly, a formula φ is satisfiable iff

$\neg\varphi$ is not a theorem of $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$: $\neg\varphi \notin \mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$, and vice versa, φ is a theorem of $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ ($\varphi \in \mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$) if $\neg\varphi$ is not satisfiable.

Using the operations \mathbf{U} and \mathbf{N} we can define all standard temporal and modal operations. For instance, modal operations may be trivially defined: $\diamond\varphi \equiv \text{true}\mathbf{U}\varphi \in \mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$; $\square\varphi \equiv \neg(\text{true}\mathbf{U}\neg\varphi) \in \mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$. The temporal operation $\mathbf{F}\varphi$ (φ holds eventually, which, in terms of modal logic, means φ is possible (denotation $\diamond\varphi$)), can be described as above: $\text{true}\mathbf{U}\varphi$. The temporal operation \mathbf{G} , where $\mathbf{G}\varphi$ means φ holds henceforth, can be defined as $\neg\mathbf{F}\neg\varphi$. Using these derived logical operations we can easily describe general knowledge operations accepted in the current framework: $\mathbf{GK}_G\varphi \equiv \square\varphi \in \mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$; $\mathbf{GK}_L\varphi \equiv \bigwedge_{1 \leq i \leq n} (\mathbf{K}_i\varphi) \in \mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$. Thus the initially specified language for $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ is a bit *superfluous* and we can omit operations for *local and global general knowledge* because they are expressible via the others.

The logic $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ evidently is more expressive compared to standard LTL and multi-agent logics in *S5*-like languages. For instance, the formula $\square\neg K_1\neg\varphi$ says that, for any future time cluster and for any state a of this cluster a state, where φ is true is *detectable* for agent 1: agent 1 has access to a state b where φ holds. The new temporal operations \mathbf{U}_s and \mathbf{U}_w bring new unique features to the language. For instance the formula $\square_w\varphi := \neg(\top\mathbf{U}_s\neg\varphi)$ codes *weak necessity*, it says that in any future time cluster $C(i)$ there is a state where φ is true. The formula $\neg(\varphi\mathbf{U}_w\square\varphi) \wedge \diamond\square\varphi$ says that, there is a future time point i , where φ holds in all future states since i , but before i φ is false in all states of some future time cluster for the current one. Such properties are problematic to express with standard modal or temporal operations. Thus, the logic $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ is expressive and interesting, and we devote the rest of the paper to finding an algorithm for verifying satisfiability in $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ in order to prove that $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ is decidable. This is not a trivial task because

Theorem 1. $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ does not have the finite model property.

This follows from the fact that the standard temporal logic $\mathcal{L}(\mathcal{Z})$ of integer numbers is a fragment of $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$, and a result of R.A. Bull (1969), cf. [45], showing that $\mathcal{L}(\mathcal{Z})$ does not have the finite model property (a proof is also given in Rybakov [22]). In [22] it is shown that the formula $\varphi_0 := \neg[\neg q \wedge \square_+\diamond_+(p \wedge \square_+q) \wedge \square_+\diamond_+(\neg p \wedge \square_+q) \wedge \square_-\diamond_-(p \wedge \square_-q) \wedge \square_-\diamond_-(\neg p \wedge \square_-q)]$ is not a theorem of $\mathcal{L}(\mathcal{Z})$, but φ_0 cannot be refuted by any finite $\mathcal{L}(\mathcal{Z})$ -frame.

3 Main Results, Decidability Algorithm

The basic technique we use is based on the reduction of formulas in the language of $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z},\mathcal{U}}$ to special inference rules and the verification of the validity these rules in frames \mathcal{Z}_C . This aims to implicitly model non-nested universal modality, which will be a useful instrument and to consider only rules (to which formulas are reduced) with non-nested non-Boolean logical operations (this simplifies the proofs and avoids the necessity to consider nested operations, and hence proofs

by induction over formula complexity). This approach combines (i) techniques to handle inference rules from [17] – [25] (where [25] solves decidability of LTL w.r.t. admissibility and again decidability of LTL itself) and (ii) techniques for a hybrid of LTL and usual knowledge logic with autonomous agents (V. Rybakov, workshop on Hybrid Logics, 2007, Dublin).

A (sequential) (inference) rule is a relation $\mathbf{r} := \frac{\varphi_1(x_1, \dots, x_n), \dots, \varphi_l(x_1, \dots, x_n)}{\psi(x_1, \dots, x_n)}$, where $\varphi_1(x_1, \dots, x_n), \dots, \varphi_l(x_1, \dots, x_n)$ and $\psi(x_1, \dots, x_n)$ are formulas constructed out of letters x_1, \dots, x_n . The letters x_1, \dots, x_n are the variables of \mathbf{r} , we use the notation $x_i \in Var(\mathbf{r})$.

Definition 2. A rule \mathbf{r} is said to be **valid** in a Kripke model $\langle \mathcal{Z}_C, V \rangle$ (notation $\mathcal{Z}_C \Vdash_V \mathbf{r}$) if $[\forall a ((\mathcal{Z}_C, a) \Vdash_V \bigwedge_{1 \leq i \leq l} \varphi_i)] \Rightarrow \forall a ((\mathcal{Z}_C, a) \Vdash_V \psi)$. Otherwise we say \mathbf{r} is **refuted** in \mathcal{Z}_C , or **refuted in \mathcal{Z}_C by V** , and write $\mathcal{Z}_C \not\Vdash_V \mathbf{r}$. A rule \mathbf{r} is **valid** in a frame \mathcal{Z}_C (notation $\mathcal{Z}_C \Vdash \mathbf{r}$) if, for any valuation V , $\mathcal{Z}_C \Vdash_V \mathbf{r}$.

For any formula φ we can convert it into the rule $x \rightarrow x/\varphi$ and employ a technique of reduced normal forms for inference rules as follows. Evidently,

Lemma 1. A formula φ is a theorem of $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z}, \mathcal{U}}$ iff the rule $(x \rightarrow x/\varphi)$ is valid in any frame \mathcal{Z}_C .

A rule \mathbf{r} is said to be in *reduced normal form* if $\mathbf{r} = \varepsilon/x_1$ where

$$\varepsilon := \bigvee_{1 \leq j \leq l} \left(\bigwedge_{1 \leq i, k \leq n, i \neq k} [x_i^{t(j,i,0)} \wedge (\mathbf{N}x_i)^{t(j,i,1)} \wedge (\mathbf{N}^{-1}x_i)^{t(j,i,2)} \wedge \right.$$

$$(x_i \mathbf{U} x_k)^{t(j,i,k,0)} \wedge (x_i \mathbf{U}_w x_k)^{t(j,i,k,1)} \wedge (x_i \mathbf{U}_s x_k)^{t(j,i,k,2)} \wedge$$

$$(x_i \mathbf{S} x_k)^{t(j,i,k,3)} \wedge (x_i \mathbf{S}_w x_k)^{t(j,i,k,4)} \wedge (x_i \mathbf{S}_s x_k)^{t(j,i,k,5)} \wedge$$

$$\left. \bigwedge_{1 \leq q \leq m} (\neg \mathbf{K}_q \neg x_i)^{t(j,i,q,6)} \wedge \mathbf{KnI} x_i^{t(j,i,3)} \right],$$

all x_s are certain letters (variables), $t(j, i, z), t(j, i, k, z) \in \{0, 1\}$ and, for any formula α above, $\alpha^0 := \alpha$, $\alpha^1 := \neg \alpha$.

Definition 3. Given a rule \mathbf{r}_{nf} in reduced normal form, \mathbf{r}_{nf} is said to be a *normal reduced form* for a rule \mathbf{r} iff, for any frame \mathcal{Z}_C , $\mathcal{Z}_C \Vdash \mathbf{r} \Leftrightarrow \mathcal{Z}_C \Vdash \mathbf{r}_{\text{nf}}$.

By the technique as for Lemma 3.1.3 and Theorem 3.1.11 in [20] we obtain

Theorem 2. There exists an algorithm running in (single) exponential time, which, for any given rule \mathbf{r} , constructs its normal reduced form \mathbf{r}_{nf} .

Decidability of $\mathcal{LT}\mathcal{L}_{IA}^{\mathcal{Z}, \mathcal{U}}$ will follow (by Lemma 1) if we find an algorithm recognizing rules in reduced normal form which are valid in all frames \mathcal{Z}_C . The starting point to handle interactions of agents is

Lemma 2. A rule \mathbf{r}_{nf} in reduced normal form is refuted in a frame \mathcal{Z}_C if and only if \mathbf{r}_{nf} can be refuted in a frame with time clusters of size square exponential from \mathbf{r}_{nf} .

For any frame \mathcal{Z}_C and some integer numbers k_1, m_1, k_2, m_2 , where $m_2 > k_2 > k_1 + 3, k_1 > m_1$ we construct the frame $\mathcal{Z}_C(k_1, m_1, k_2, m_2)$ from \mathcal{Z}_C as follows. $\mathcal{Z}_C(k_1, m_1, k_2, m_2) := \langle \bigcup_{m_1 \leq i \leq m_2} C(i), R, R_1, \dots, R_m, \text{Next} \rangle$, where R is the accessibility relation from \mathcal{Z}_C extended by pairs (x, y) , where $x \in C(i), y \in C(j)$ and $i, j \in [m_1, k_1]$, or $i, j \in [k_2, m_2]$. Any relation R_j is simply transferred from \mathcal{Z}_C , and Next and Prev are taken from \mathcal{Z}_C and extended by $\forall a \in C(m_2) \forall b \in C(k_2) (a \text{ Next } b = \text{true})$; $\forall a \in C(m_2) \forall b \in C(k_2) (b \text{ Prev } a = \text{true})$; $\forall a \in C(m_1) \forall b \in C(k_1) (a \text{ Prev } b = \text{true})$; $\forall a \in C(m_1) \forall b \in C(k_1) (b \text{ Next } a = \text{true})$. For any valuation V of letters from a formula φ in $\mathcal{Z}_C(k_1, m_1, k_2, m_2)$ the truth value of φ can be defined at elements of $\mathcal{Z}_C(k_1, m_1, k_2, m_2)$ by the rules similar to the ones given for the frames \mathcal{Z}_C above (just in accordance with the meaning of logical operations). Due to limitations on the length of the paper we omit a detail description of these rules. Using Lemma 2 as the basis, we can derive

Lemma 3. A rule \mathbf{r}_{nf} in reduced normal form is refuted in a frame \mathcal{Z}_C iff \mathbf{r}_{nf} can be refuted in a frame $\mathcal{Z}_C(k_1, m_1, k_2, m_2)$ by a valuation V of special kind, where the size of the frame $\mathcal{Z}_C(k_1, m_1, k_2, m_2)$ is triple exponential in \mathbf{r}_{nf} .

We do not specify in the formulation of this lemma properties required for the valuation V , but they are essential since any frame $\mathcal{Z}_C(k_1, m_1, k_2, m_2)$ is not an $\mathcal{LTL}_{IA}^{\mathcal{Z}, \mathcal{U}}$ -frame (cf. $\mathcal{LTL}_{IA}^{\mathcal{Z}, \mathcal{U}}$ does not have the finite model property). From Theorem 2, Lemma 1 and Lemma 3 we derive

Theorem 3. The logic $\mathcal{LTL}_{IA}^{\mathcal{Z}, \mathcal{U}}$ is decidable. The algorithm for checking a formula to be a theorem of $\mathcal{LTL}_{IA}^{\mathcal{Z}, \mathcal{U}}$ consists in verification of validity rules in reduced normal form at frames $\mathcal{Z}_C(k_1, m_1, k_2, m_2)$ of size triple-exponential in the size of reduced normal forms w.r.t. valuations of special kind.

It is possible also to apply the technique from this paper to weakened versions of the logic $\mathcal{LTL}_{IA}^{\mathcal{Z}, \mathcal{U}}$, say with omitted strong or weak versions of the operations **U** or **S**, with omitted **N** or **N⁻¹** and to obtain similar results about decidability. Also some restrictions for agents accessibility relations R_i may be considered by the introduction of a hierarchy between these relations R_i .

4 Conclusion, Future Work

The paper develops a technique for proving decidability of the logic $\mathcal{LTL}_{IA}^{\mathcal{Z}, \mathcal{U}}$ and a number of similar logics. The suggested approach is proven to be flexible enough to work with a variety of logics from AI and CS. There are many prospective avenues of research on logic $\mathcal{LTL}_{IA}^{\mathcal{Z}, \mathcal{U}}$ and its variants. For instance, the next interesting candidate for the research is a variant of the logic with a language representing a *hierarchy* of interacting agents.

Besides, interesting question is whether it is possible to extend the methods of this paper to handle the case of hybrid non-linear temporal logics (e.g. branching time logics, $S4_T$, $K4_T$) with interacting agents. An open question is the problem of axiomatizability. Another interesting problem concerns complexity issues and possible ways of refining the complexity bounds in the algorithm. Problems of decidability w.r.t. admissible inference rules in fusions of LTL based at \mathcal{N} (without Since and Previous) and multi-modal logics with interacting agents are not investigated yet. The problem of describing bases for rules admissible in such logics is also open to date.

References

1. Hakansson, A., Hartung, R.L.: Autonomously creating a hierarchy of intelligent agents using clustering in a multi-agent system. In: Proc. of the 2008 Int. Conf. on Artificial Intelligence, IC-AI 2008, pp. 89–95 (2008)
2. Apelkrans, M., Håkansson, A.: Applying Multi-Agent System Technique to Production Planning in Order to Automate Decisions. In: Håkansson, A., Nguyen, N.T., Hartung, R.L., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2009. LNCS, vol. 5559, pp. 193–202. Springer, Heidelberg (2009)
3. Barringer, H., Fisher, M., Gabbay, D., Gough, G.: Advances in Temporal Logic. Applied logic series, vol. 16. Kluwer Academic Publishers, Dordrecht (1999)
4. Bull, R.A.: An Algebraic Study of Tense Logics With Linear Time. *The Journal of Symbolic Logic* 33, 27–38 (1968)
5. Bull, R.A.: Note on a Paper in Tense Logic. *The Journal of Symbolic Logic* 34, 215–218 (1969)
6. Clarke, E., Grumberg, O., Hamaguchi, K.P.: Another look at LTL Model Checking. In: Dill, D.L. (ed.) CAV 1994. LNCS, vol. 818, pp. 415–427. Springer, Heidelberg (1994)
7. Daniele, M., Giunchiglia, F., Vardi, M.: Improved Automata Generation for Linear Temporal Logic. In: Halbwachs, N., Peled, D.A. (eds.) CAV 1999. LNCS, vol. 1633, pp. 249–260. Springer, Heidelberg (1999)
8. de Jongh, D., Veltman, F., Verbrugge, R.: Completeness by construction for tense logics of linear time. In: Troelstra, A.S., Visser, A., van Benthem, J.F.A.K., Veltman, F.J.M.M. (eds.) Liber Amicorum for Dick de Jongh. Institute of Logic, Language and Computation, Amsterdam (2004), <http://www.illc.uva.nl/D65/>
9. Fagin, R., Halpern, J., Moses, Y., Vardi, M.: Reasoning About Knowledge. The MIT Press, Cambridge (1995)
10. Gabbay, D.M., Hodkinson, I.M.: An axiomatisation of the temporal logic with Until and Since over the real numbers. *Journal of Logic and Computation* 1, 229–260 (1990)
11. van der Hoek, W., Wooldridge, M.: Towards a Logic of Rational Agency. *Logic Journal of the IGPL* 11(2), 133–157 (2003)
12. Halpern, J., Shore, R.: Reasoning about common knowledge with infinitely many agents. *Information and Computation* 191(1), 1–40 (2004)
13. Hodkinson, I.: Temporal Logic and Automata. In: Gabbay, D.M., Reynolds, M.A., Finger, M. (eds.) Temporal Logic: Mathematical Foundations and Computational Aspects, ch. II, vol. 2, pp. 30–72. Clarendon Press, Oxford (2000)
14. Kacprzak, M.: Undecidability of a multi-agent logic. *Fundamenta Informatiae* 45(2-3), 213–220 (2003)

15. Manna, Z., Pnueli, A.: Temporal Verification of Reactive Systems: Safety. Springer, Heidelberg (1995)
16. Pnueli, A.: The Temporal Logic of Programs. In: Proc. of the 18th Annual Symp. on Foundations of Computer Science, pp. 46–57. IEEE, Los Alamitos (1977)
17. Rybakov, V.V.: A Criterion for Admissibility of Rules in the Modal System $S4$ and the Intuitionistic Logic. Algebra and Logic 23(5), 369–384 (1984) (Engl. Translation)
18. Rybakov, V.V.: Rules of Inference with Parameters for Intuitionistic logic. Journal of Symbolic Logic 57(3), 912–923 (1992)
19. Rybakov, V.V.: Hereditarily Structurally Complete Modal Logics. Journal of Symbolic Logic 60(1), 266–288 (1995)
20. Rybakov, V.V.: Admissible Logical Inference Rules. Series: Studies in Logic and the Foundations of Mathematics, vol. 136. Elsevier Sci. Publ., North-Holland (1997)
21. Rybakov, V.V.: Construction of an Explicit Basis for Rules Admissible in Modal System $S4$. Mathematical Logic Quarterly 47(4), 441–451 (2001)
22. Rybakov, V.V.: Logical Consecutions in Discrete Linear Temporal Logic. Journal of Symbolic Logic 70(4), 1137–1149 (2005)
23. Rybakov, V.V.: Logical Consecutions in Intransitive Temporal Linear Logic of Finite Intervals. Journal of Logic Computation 15(5), 633–657 (2005)
24. Rybakov, V.: Until-Since Temporal logic Based on Parallel Time with Common Past. In: Artemov, S., Nerode, A. (eds.) LFCS 2007. LNCS, vol. 4514, pp. 486–497. Springer, Heidelberg (2007)
25. Rybakov, V.: Linear Temporal Logic with Until and Next, Logical Consecutions. Annals of Pure and Applied Logic 155(1), 32–45 (2008)
26. Babenyshev, S., Rybakov, V.V.: Logic of Discovery and Knowledge: Decision Algorithm. In: Lovrek, I., Howlett, R.J., Jain, L.C. (eds.) KES 2008, Part II. LNCS (LNAI), vol. 5178, pp. 711–718. Springer, Heidelberg (2008)
27. Rybakov, V.V.: Linear Temporal Logic LTL_k extended by Multi-Agent Logic K_n with Interacting Agents. J. Log. Comput. 19(6), 989–1017 (2009)
28. Babenyshev, S., Rybakov, V.V.: Temporal Logic for Modeling Discovery and Logical Uncertainty. In: Velásquez, J.D., Ríos, S.A., Howlett, R.J., Jain, L.C. (eds.) KES 2009. LNCS, vol. 5712, pp. 16–23. Springer, Heidelberg (2009)
29. Babenyshev, S., Rybakov, V.V.: Describing Evolutions of Multi-Agent Systems. In: Velásquez, J.D., Ríos, S.A., Howlett, R.J., Jain, L.C. (eds.) KES 2009. LNCS, vol. 5711, pp. 38–45. Springer, Heidelberg (2009)
30. Babenyshev, S., Rybakov, V.V.: A Framework to Compute Inference Rules Valid in Agents' Temporal Logics. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010. LNCS, vol. 6276, pp. 230–239. Springer, Heidelberg (2010)
31. Babenyshev, S., Rybakov, V.V.: Multi-agent Logics with Interacting Agents Based on Linear Temporal Logic: Deciding Algorithms. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2010. LNCS, vol. 6114, pp. 337–344. Springer, Heidelberg (2010)
32. van Benthem, J., Bergstra, J.A.: Logic of Transition Systems. Journal of Logic, Language and Information 3(4), 247–283 (1994)
33. Vardi, M.: An automata-theoretic approach to linear temporal logic. In: Proceedings of the Banff Workshop on Knowledge Acquisition, Banff 1994 (1994)
34. Vardi, M.: Reasoning about the past with two-way automata. In: Larsen, K.G., Skyum, S., Winskel, G. (eds.) ICALP 1998. LNCS, vol. 1443, pp. 628–641. Springer, Heidelberg (1998)

KAMET II: KAMET Plus Knowledge Generation

Osvaldo Cairó and Silvia Guardati

Department of Computer Science, ITAM,
Río Hondo 1, 01080 México DF, México
cairo@itam.mx, guardati@itam.mx

Abstract. The knowledge acquisition (KA) process has evolved during the last years. Today KA is considered a cognitive process that involves both a dynamic modeling and knowledge generation activities. This should be seen as a spiral of epistemological and ontological content that grows up by transforming tacit knowledge into explicit knowledge, which in turn becomes the basis for a new spiral of knowledge generation. This paper shows some of our attempts to build a new knowledge acquisition methodology that collects and includes all of these ideas. KAMET II, the evolution of KAMET [1], represents a modern approach for building diagnosis-specialized knowledge models that could be run by Protégé.

Keywords: Knowledge acquisition, knowledge modeling, knowledge generation.

1 Introduction

Knowledge Acquisition started as an attempt to solve the main bottleneck in developing knowledge-based systems (KBS). Thousands of KBS have been developed and applied world-wide in different knowledge domains. Although, technologies have been improved in recent years, knowledge acquisition still remains the main factor that hamper a well controlled KBS life cycle.

Knowledge undoubtedly represents the main competitive advantage of an organization. The idea is simple: *apply knowledge to a work environment in order to create value*. Knowledge is strongly related to intangible resources, intellectual capital, and market assets of an organization. Despite its importance, we are still far from understanding the process in which an organization creates and utilizes knowledge.

While the problem is clear, the solution is hard to implement. Knowledge is a fluid mix of framed experience, values, expertise, contextual information and insight that provides a suitable environment and structure for evaluating and incorporating new information and experiences. The theoretical or practical understanding of a subject is the individual's ability of how to get something done, but knowledge is often tacit; that is, it lies in the mind of individuals and therefore it is difficult to transfer to another person by means of writing it down or verbalizing it. Although much is known about neural and biochemical activities, little is known about memory and thinking. The process whereby humans represent knowledge is not very clear yet [2]. Efforts to acquire and model the *know-how*, the *know-why* and the *care-why* of an expert must undoubtedly involves knowledge and ideas from different areas, such as psychology, sociology, philosophy and computer science.

This paper addresses this important problem. We conceive the process of knowledge acquisition as a cognitive process that involves both a dynamic modeling process and a knowledge generation process. These processes are integrated in a spiral of epistemological and ontological content that grows up by transforming tacit knowledge into explicit, which becomes the basis for a new spiral of knowledge generation. These processes involve deduction, induction, creativity, efficiency [3]. Epistemology is also very important because it concerns with the nature and scope of knowledge. On the other hand, ontology is as well fundamental by two main reasons: it relates with human nature, existence and properties of mind, and the formal representation of knowledge as a set of concepts within a domain, and the relationships between those concepts.

We take into account the previously aforementioned and thus develop KAMET II, an evolution of KAMET. It is a methodology based on models designed to manage knowledge acquisition from multiple knowledge sources (KS). It provides a strong mechanism with which to achieve KA in an incremental fashion, in a cooperative environment, and in a shared context for knowledge creation. KAMET II represents a modern approach for building diagnosis-specialized knowledge models that could be run by Protégé. KAMET II seeks to be general, although it is mainly directed toward problems of diagnosis.

2 KAMET II Life-Cycle Model: The Knowledge Generation Process

The KAMET II life-cycle model (LCM) provides a graphical framework for managing the knowledge acquisition process. The graphical framework also helps to set up and facilitate ways to characterize and organize the knowledge acquired from multiple knowledge sources, to share knowledge, implement the required actions, review the project situation, identify risks, monitor project progress, and check the quality. Besides providing structure to avoid problems of corporate IT bureaucracy, much of the motivation behind utilizing a life cycle model as a knowledge generation process is based on the search for the efficient transformation of tacit knowledge into explicit knowledge. We are much more interested in the dynamic process of knowledge generation than the stockpiling of knowledge. This is one of the main differences with the version of the methodology. KAMET emphasized the spiral model and the essence of cooperative work –two interesting concepts that allow risk reduction, but which did not specifically focus on the knowledge generation process.

The most important features of the KAMET II life cycle are: spiral structured, results-driven, risk-driven, scalable, extensible, and quality control [1]. The KAMET II life cycle consists of four stages: *the strategic planning of the project*, *initial model building*, *feedback model building*, and *final model building*. Each stage involves a process of knowledge transformation. Following is a brief description of different stages.

2.1 The Strategic Planning of the Project

The first stage, the strategic planning of the project, is essential for the development of the project. The Project Manager (PM) and the four groups involved in the project (Knowledge Engineering (KE), Human Experts (HE), representatives of potential users (PU), and fund sponsors (FS)) have to interact and must be in total agreement with the definition of the project to assure its success. This is the *socialization stage* in which ideas, views, experiences and knowledge should be shared through face-to-face interactions. This process is necessarily context-specific in terms of who participates and how they participate. Social, cultural and historical contexts are important for human beings, as such context provide the basis for interpreting information to create meaning [4].

The following are the steps comprised in the first stage [1]: a) Define project goals, b) Identify potential users, c) Specify potential benefits, d) Divide the knowledge domain into sub-domains, e) Identify the knowledge sources, f) Definition of model verification and validation mechanisms, g) Build the project's dictionary, h) Specify other necessary resources to attain KA, i) Define techniques to achieve knowledge acquisition, j) Estimate time to complete the knowledge acquisition stage, k) Estimate project costs, and l) Specify project documentation.

2.2 Initial Model Building

The *externalization process* takes place in the second stage. It is the time for transforming tacit knowledge into explicit knowledge. When tacit knowledge is made explicit, knowledge is crystallized. This means knowledge can be shared by others and therefore, becomes the basis for a new process of knowledge generation.

In this stage, the KE elicits knowledge from different KS and proceed to build the initial model, which is constituted by one or more models as we will explain later. This stage involves the largest number of risks, which mainly arise because interviews involve introspection and verbal expression of knowledge, resulting in a difficult task for humans, and especially for experts. On the other hand, if the communication language among PM, KE and HE is not clear, this may also cause conflicts. The success of the initial model is heavily dependent on the skills of KEs to socialize with the experts and to formalize the tacit knowledge.

Steps comprised in the second stage are the following: a) Attain knowledge elicitation from multiple knowledge sources, b) Reassessing the project time, c) Develop a library of cases, d) Develop the initial model, e) Verification and validation of the initial model, and f) Documentation of the initial model.

2.3 Feedback Model Building

It is the time for *combination* -- the process of converting explicit knowledge into more complex and systematic sets of explicit knowledge [4]. The KE distributes the initial model among the different knowledge sources to be analyzed. Ideas, experiences or perspectives are exchanged in relation to the model. Because individuals typically have different views, training, ideas, knowledge and experience, it is logical that differences are common and inevitable at this time. This should not be a cause for

concern. The synthesis of these differences should be used to generate new knowledge and bring forth diverse views in reference to the created artifacts.

Finally, the PM and the KE, jointly with the experts, review and analyze the changes introduced to the initial model and constructs the feedback model. The inaccuracy of the model at the end of this stage must be less, since the model now was enriched and expresses the knowledge and experience of several specialists in the knowledge domain of the application. It must be remembered that the feedback model is only a refined and better initial model. Following are the steps that constitute the third stage: a) Distribute the initial model among experts, b) Analysis by experts of the initial model, c) Develop the feedback model incorporating the different views of experts, d) Verification and validation of the feedback model, and e) Documentation of the feedback model.

2.4 Final Model Building

In the last stage, the multiple KSs participate in a series of interviews, under the coordination of the PM, to develop the final model. The stage is considered to be over when the model satisfies the proposed objectives with a high degree of plausibility and/or there are no experts capable of further transforming it. Inaccuracy at the end of this stage must be minimal, since the model now expresses the knowledge acquired from multiple KSs, which collaborated in different degrees and ways to solve the problem. The final model shows that explicit knowledge can be redistributed among team members and converted into tacit knowledge again. Following are the steps that constitute the fourth stage: a) Analysis of the feedback model by the experts, b) Develop the final model incorporating new and more specific opinion from the experts, c) Verification and validation of the final model, and d) Documentation of the final model.

3 The KAMET II Conceptual Modeling Language (CML)

It is worth beginning this section by raising the following question: *Can a good language for knowledge modeling be formulated?* The answer definitively is yes, but research evidence shows that this language has not yet been developed. There are several reasons why, but perhaps the main one is that it has not really been necessary until now. We think it is time to address the issue.

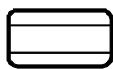
The transfer of knowledge directly from different knowledge sources to artificial machines is less organized, reliable, comprehensible, and effective than when it is represented in intermediate models. We require methods for the construction of models for the acquisition of knowledge that are much more effective. We need a method that allows us to analyze and comment about the world.

A good knowledge modeling language should provide an extensive vocabulary in which the knowledge can be expressed and modeled in such a way that allows, for instance, the understanding and the reasoning by means of visual illustration or representation. This implies the possibility of understanding of concepts and ideas, visualized through knowledge models without using linguistic or algebraic means. The idea is not new. It was introduced by Leibniz many years ago.

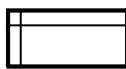
The KAMET II modeling language takes into account the above-mentioned points and attempts to provide the necessary elements for KE to build effective models. Up to now, the results will depend on the judgment of knowledge engineers, as well as on the logical, psychological and epistemological considerations that they make at the right time. We are aware that KAMET II does not represent the ideal solution, but we also believe that shows the way forward for the next few years. It is also important to remember that KAMET II seeks to be general, although it is mainly directed toward problems of diagnosis.

3.1 The KAMET II CML Assumptions

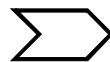
The KAMET II CML has three levels of abstraction. The first one corresponds to *structural constructors* and *structural components*. They are used primarily to highlight the problem itself. We distinguish between problem, classification and subdivision (Fig. 1).



Problem: It refers to a situation, condition, or issue that is yet unresolved.



Classification: A characteristic inference structure that systematically relates data to a preenumerated set of solutions by abstraction, heuristic association, and refinement.



Subdivision: It is the act of dividing a problem into small pieces that are easier to solve.

Fig. 1. Structural constructors

The structural components (Fig. 2) are used to establish the characteristics and possible solutions of the problem. We distinguish among symptoms, antecedents, time, value, inaccurate, process, formula, solution and examination.

The second level of abstraction corresponds to *nodes* (N) and *composition rules* (CR). Nodes are built using structural constructors and structural components. We distinguish between three different types of nodes: *initial*, *intermediate* and *terminal*. On the other hand, composition rules (Fig. 3) are the ones that permit the adequate combination of nodes. The third level of abstraction corresponds to the *global model*. It consists of at least one initial node, any number of intermediate nodes, and one or more terminal nodes. A global model should represent the knowledge acquired from multiple knowledge sources in a specific knowledge domain.

3.2 The KAMET II CML Formalization

The formalization of KAMET II CML is based more on a metalanguage, than on a strict group of theorems and mathematical proofs. The characterization of the method through diagrammatic conventions and postulates can be summarized as follows.

		
Symptom: Indicates the presence of an alteration, disorder or abnormality.	Antecedent: It shows a preceding event, condition, or cause. It might refer to the previous circumstances that can be used to judge something that might happen.	Solution: Refers to an explanation or answer to the problem.
		
Time: Expresses a one-dimensional quantity used to sequence events, to quantify the durations of events and the intervals between them.	Value: Concerning the value of symptoms, antecedents, groups, etc.	Inaccurate: Used to indicate a loss of precision or accuracy.
		
Process: Describes the act of taking something through an established and usually routine set of procedures to convert it from one form to another	Formula: It is a set form of words in which something is defined, stated, or declared .	Examination: Expresses a recommendation or necessity of making studies, exams, tests to determine an alteration, disorder or abnormality.

Fig. 2. Structural components

			
Subdivision: Shows a subdivision.	Implication: It represents a connection from a source to a complication.	Action: Expresses that something must be completed: a formula, an examination, etc.	Union: The line shows a connection between subdivisions.

Fig. 3. Composition rules

3.3 Diagrammatic Conventions

A diagrammatic convention is mainly a chart, graph, drawing or outline designed to demonstrate or explain how something works or to clarify the relationship between the parts of a whole. Following are the diagrammatic conventions of the method:

- DG1.** The structural constructors and structural components can be *named* using a numerical or linguistic label. The use of names accelerates and facilitates the construction of models.
- DG2.** The *indicator* is used to set the number of elements that must be present in either a structural component or group. It is represented with a square and is located in the upper right-hand corner of the group or the structural component. An indicator is named in three different ways: a n is used to express the exact number of elements that must be present, a $n+$ is used to indicate that at least n elements must be present, and n, m is used to show the minimum and maximum number of elements that must be present, where $m > n$.

- DG3.** A *chain* is defined as the link of two or more symptoms, antecedents, and/or groups (DG4). The order of the link is irrelevant.
- DG4.** A *group* is defined as a special chain. The linked elements have times and/or values in common, or are related among them through an indicator. The group concept is recursive.
- DG5.** *Assignment* is defined as the process of labeling a node. The objective of the assignment is to be able to reuse the node in any other part of the model without having to redefine it. It allows reusing a complete node not only in form but also in content. Reusing is a universal principle to cope with complexity and to avoid redesigning or redeveloping parts of a product, which already exist. The assignment provides greater flexibility in modeling.

3.4 Postulates

The *postulate* or axiom is a proposition in logic that is not proved or demonstrated but considered to be either self-evident or assumed as true as a basis for reasoning. Its truth is taken for granted, and serves as a starting point for deducing and inferring other proposition. Following are the postulates of the method:

- P1.** The structural component *time* should always be placed to the right of a group, problem, subdivision, antecedent, symptom, etc.
- P2.** The structural component *value* is always placed above a symptom, antecedent or group. The value component can make use of an indicator.
- P3.** The solution component is only related to structural constructors.
- P4.** There are three types of nodes: initial (I), intermediate (M) and terminal (T).
- P5.** The *nodes* are related using composition rules. The following relationships are possible: initial with terminal, initial with intermediate, intermediate with intermediate, and intermediate with terminal.
- P6.** An *initial node* represents a symptom, antecedent, group, or chain. It is used to describe a part of the problem. It does not have input flow and can have more than one outflow.
- P7.** The *intermediate node* is used to describe an intermediate part of the problem. It may have one or more inflows and one or more outflows.
- P8.** A *terminal node* represents a structural constructor. It has one or more input flows. The outflows are only used to show possible solutions.
- P9.** The initial and intermediate nodes can be grouped together, without losing their properties or functions, into molecular nodes. These nodes, in turn, will act as a node in their own right. The molecular nodes are formed through conjunctions or disjunctions.
- P10.** The *composition rules* are used to relate mainly the different nodes and the structural components with the solution component.

3.5 A Simple Example of Modeling

In this section we will provide a simple example of modeling in order to illustrate the methodology sketched in the previous section. The example concerns diagnosing faults in electricity (figure 4). The model expresses that the problem P1 can occur due

to two different situations. In the first one, the model expresses that if the symptoms 1 and 5 are known to be true then we can deduce the problem P1 is true with probability 0.6. In the second one, the model shows that if symptoms 1 and 2 are observed then we can conclude that the problem is P1 with probability 0.70. On the other hand, we can deduce that the problem P3 is true with probability 0.40 if symptoms 3 and 4 are known to be true. Finally, we can reach a conclusion that the problem is P2 with probability 0.90 if problems P1 and P3 and the symptom 7 are observed.

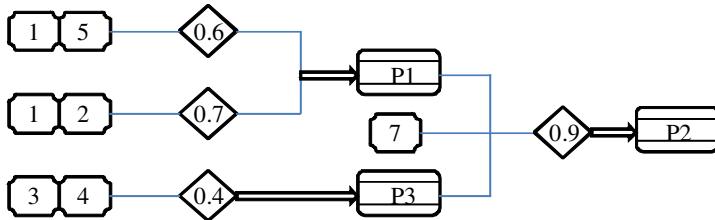


Fig. 4. Simple electrical diagnosis

The next example is from a KBS for the diagnosis of headache disorders, cranial neuralgia's, and facial pain.

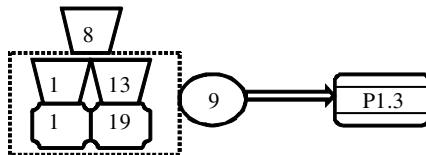


Fig. 5. Ophthalmoplegic migraine

4 KAMET II Architecture

The KAMET II methodology relies on a conceptual and formal framework for the specification of knowledge-based systems. The conceptual framework is a typical, well-known *four component architecture* that defines different elements to solve diagnosis problems: a *task* that defines the problem that should be solved by the knowledge-based systems, a *problem-solving method* that defines the reasoning process of the knowledge-based systems, and a *domain model* that describes the domain knowledge of the KBS. Each of these elements is described independently to enable the reuse of each of them. Additionally, a fourth element, known as *adapter*, is introduced to adjust the three other independent and reusable parts.

Problem-solving methods (PSMs) are ready-made software components that can be assembled with domain knowledge-bases to create application systems. The PSM should take into account both the reasoning process and the knowledge needed to solve the tasks. The domain model, on the other hand, introduces the domain knowledge and is represented by ontology. The use of ontologies in constructing a KBS is pervasive. They support the modeling of the domain-knowledge counterpart of PSMs

in knowledge applications. However, PSMs and domain ontologies are developed independently and therefore need to be reconciled to form a coherent knowledge system. As the basis for reconciliation, PSMs choose the format and semantics of the knowledge that they expect from the domain to perform their task [12]. Finally, the adapter allows the combination of the three other components that may differ in syntactical input and output descriptions.

5 KAMET II Methodology: Applications and Results

KAMET II presents a different way of doing things and a collection of new ideas; let us ultimately build more robust models of knowledge, and building knowledge-based systems more efficient. It is undoubtedly very useful. However, the main motivation behind this work is to introduce the knowledge acquisition as a cognitive process, as a spiral of epistemological and ontological content that grows up by transforming tacit knowledge in explicit knowledge. The way of doing things, although they did not involve a fundamental change in the structure of the methodology, necessarily imply a change of mind, a change of principles and strategies, and a fundamental change in the way of seeing the problem.

The KAMET Methodology has been successfully used in different applications and knowledge domains. We have developed several KBS using KAMET mainly in medicine –cranial neuralgias and uveitis-, telecommunications, recruiting, concrete design, scheduling, human resources management system, customer services, etc. A great deal of literature has also appeared on KAMET in recent years ([5], [6], [7], [8], [9], [10], [11]). We do think KAMET II, which involves a new dynamic modeling process and a revolutionary knowledge generation process, provides the necessary elements for KE to build KBS. The methodology also focuses naturally on risk-reduction, which is a fundamental part in software and knowledge engineering.

6 Conclusions

In this paper, we presented a renewed and fresh knowledge acquisition methodology from multiple knowledge sources. It is worth noting there are two main goals in developing KAMET II. The first one is to improve the phase of knowledge acquisition making it efficient. The second, and more important, is to introduce the knowledge acquisition as a cognitive process, as a spiral of epistemological and ontological content that grows upward by transforming tacit knowledge into explicit knowledge.

We would like to close this paper by stating that this is one of the first attempts at incorporating both a dynamic modeling process and a knowledge generation process in a knowledge acquisition methodology. We also know that much remains to be done to quantify the effects we have pointed out. We also think that the principles illustrated here may have a wider applicability, and should be employed in a more general manner in order to provide a deeper understanding of knowledge acquisition.

Acknowledgements. This work has been founded by Asociación Mexicana de Cultura A.C.

References

1. Cairó, O.: KAMET: A Comprehensive Methodology for Knowledge Acquisition from Multiple Knowledge Sources. *Expert Systems with Applications* 14, 1–16 (1998)
2. Vámos, T.: Expert Systems and the Ontology of Knowledge Representation. In: Lee, J., Liebowitz, J., Chae, Y. (eds.) *Critical Technology, Cognizant Communication Corporation*, pp. 3–12 (1996)
3. Nonaka, I., Toyama, R.: The knowledge-creating theory revisited: knowledge creation as a synthesizing process. *Knowledge Management Research & Practice* 1, 2–10 (2003)
4. Nonaka, I., Toyama, R., Konno, N.: SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation. *Long Range Planning* 33, 5–34 (2000)
5. Chen, J., Hwang, G., Hwang, G., Chu, C.: Analyzing Domain Expertise by Considering Variants of Knowledge in Multiple Time Scales. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) *KES 2005. LNCS (LNAI)*, vol. 3683, pp. 1324–1330. Springer, Heidelberg (2005)
6. Hwang, G., Chen, J., Hwang, G., Chu, H.: A time scale-oriented approach for building medical expert systems. *Expert System with Applications* 31(2), 299–308 (2006)
7. Chu, H., Hwang, G.: A Delphi-based approach to developing expert systems with the co-operation of multiple experts. *Expert System with Applications* 34(4), 2826–2840 (2008)
8. Elfaldil, N.: Knowledge extraction from rise-time auto-correlated patterns. *International Journal of Information Acquisition* 5(2), 181–187 (2008)
9. Lin, S., Tseng, S., Teng, C.: Dynamic EMCUD for knowledge acquisition. *Expert System with Applications* 34(2), 833–844 (2008)
10. Beydoun, G., Tran, N., Low, G., Henderson-Sellers, B.: Foundations of Ontology-Based MAS Methodologies. In: Kolp, M., Bresciani, P., Henderson-Sellers, B., Winikoff, M. (eds.) *AOIS 2005. LNCS (LNAI)*, vol. 3529, pp. 111–123. Springer, Heidelberg (2006)
11. Tseng, S., Lin: SVODKA: Variant objects discovering knowledge acquisition. *Expert System with Applications* 36(2), 2433–2450 (2009)
12. Crubézy, M., Musen, M.: Ontologies in Support of Problem Solving. In: Staab, S., Studer, R. (eds.) *Handbook on Ontologies*, pp. 321–341. Springer, Heidelberg (2003)

Use of Metadata for Access Control and Version Management in RDF Database

Kazuhiro Kuwabara¹ and Shotaro Yasunaga^{2,*}

¹ Department of Information Science and Engineering, Ritsumeikan University

² Graduate School of Science and Engineering, Ritsumeikan University,

1-1-1 Noji-Higashi, Kusatsu, Shiga 525-8577 Japan

kuwabara@is.ritsumei.ac.jp

Abstract. This paper proposes using metadata in a Resource Description Framework (RDF) database for version management and access control. The proposed mechanism attaches metadata containing the version and access control information to an RDF statement. When new data (RDF statements) are derived based on the rules, the metadata are added to the new data. When a query is made to the RDF database, it is rewritten so that the metadata are considered. In addition, the mechanism is intended to be used on top of existing RDF database software for easy portability. An application of the proposed mechanism is shown in the context of the construction of a topic database for the conversation support of people with language disorders.

Keywords: RDF, metadata, access control, version management.

1 Introduction

This paper proposes a method for version management and access control for a Resource Description Framework (RDF) database. Constructing a RDF database, especially adding data (resources) and links between them is laborious. We can construct an RDF database by combining multiple RDF databases developed by different users. By linking two resources in different databases, we can easily construct a larger RDF database.

Integrating multiple databases, especially those developed by different users, involves some issues that need to be solved. One is achieving access control. Often a database is constructed by an individual for his/her own personal purpose. The user specifies which data should be shared with whom. If the user can specify such access control, the user's personal RDF database can be published and shared with other (possibly public) databases. The second issue is that database construction inevitably involves trial and error processes. The user may want to test a particular link between two resources in different databases and verify the results. The user may also want to compare different ways to make links between two databases.

* Presently with NTT DATA Corporation.

To tackle these issues, we propose a mechanism that uses metadata attached to an RDF statement (a triple) for access control and version management. The metadata for access control include the ‘owner’ of the triple and information about access permission. By introducing version management, we can also enable a user to easily compare two different versions. This mechanism is intended to be constructed on top of existing RDF database software so that the internal structure of the RDF database does not have to be modified.

This paper is structured as follows. The next section discusses related works. Then, the metadata for version management and access control are presented with an example. Finally, an application to the construction of a topic database is described, followed by a conclusion.

2 Related Works

There are many works on access control model for RDF data (for example, [14][7][8][11]). There are also access control mechanisms proposed for specific application domains. For example, access control for online photo albums was proposed based on tags attached to photos and linked data [2], where the tags are given by a user, and the linked data refer to the user’s profile based on ‘Friend-Of-A-Friend’ (FOAF). With the FOAF information, flexible access control is achieved. In addition, an access control list (ACL) described in RDF is used to represent such permission as ‘read’ or ‘write,’ that is suitable for a collaborative authoring environment [6]. The proposed mechanism here mainly targets the construction of a topic database by many users, and is intended to provide a simple access control mechanism that is useful for combining RDF databases built by different users.

For version management, OWL specification has an `owl:versionInfo` property that represents the version of the ontology in a particular file. Managing the statement-based version control is not easy in this case. Since we consider cases where a link is defined to connect different resources in multiple databases, we must be able to do version management on a statement basis.

For statement level versioning, a scheme of recording the change history of addition and removal of triples was proposed [10]. In addition, multidimensional RDF was proposed to handle multiple contexts [5]. In contrast to these approaches, the proposed mechanism of this paper is intended for use on top of an existing RDF database software, and it puts emphasis on software portability.

3 Use of Metadata

The main idea of the proposed method is adding metadata regarding access control and version management to an RDF statement (a triple). When a new triple is derived, the metadata are added based on the metadata on the RDF triples that caused a new triple to be derived. The addition of the metadata is described using rules. When a query is made to the RDF database, additional conditions are added to the original query so that only triples are retrieved

that satisfy the access control and version management conditions. From the viewpoint of the RDF database, metadata regarding access control and version management are stored in the RDF database in the same way as the original data. However, when a triple is added, the metadata are calculated and attached to the new triple. In addition, a query is rewritten to handle version management and access control. The RDF database itself is not required to be modified to handle the metadata.

To add metadata to each RDF statement, the RDF statement is reified (Fig. 1). In this example, triple (`:Cat rdfs:subClassOf :Animal`) has the metadata. The metadata for the access control are represented by `sem:accessInfo` and the version information is represented by `sem:versionInfo`. Here, prefix `sem` represents the vocabulary defined for this paper and is used throughout it¹.

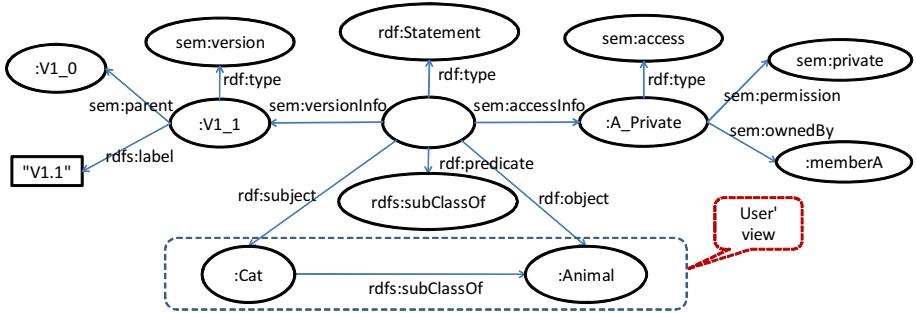


Fig. 1. Attaching metadata

3.1 Metadata for Access Control and Version Management

The metadata for access control (`sem:accessInfo`) has the information on access permission (`sem:permission`) and owner (`sem:ownedBy`). The metadata for version management (`sem:versionInfo`) denote the version in which the statement is held. The metadata may have a label that describes the version's information that is suitable for users. Moreover, the version can have a ‘parent’ version from which definitions are inherited. The parent version is specified by the `sem:parent` property.

Metadata are added in two situations. One is when a user manually adds a triple, and the other is when a new triple is derived from the existing triple(s) based on the ontology definition.

When a user adds a triple, the user must specify its access permission in addition to the version in which it exists. If no such information is given, the triple is assumed to be available to anyone and to any version, and no metadata regarding access control and version management are attached.

Figure 2 shows how a newly derived triple is attached with metadata. We assume that triple (`:Kitty rdf:type :Cat`) is originally defined and is

¹ In the implementation, `@prefix sem: <http://www.semlab.jp/#>` is assumed.

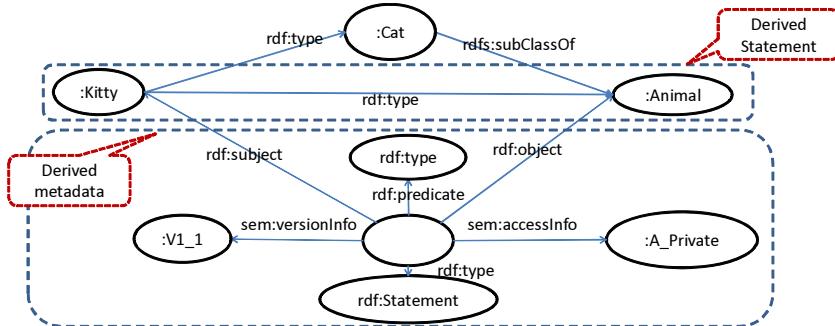


Fig. 2. Example of deriving metadata

valid in all versions and can be accessed by anyone. In addition, triple (`(:Cat rdfs:subClassOf :Animal)`) is defined in the version of `:V1_1` and created by user A and specified as only accessible by user A (as shown Fig. 2). The definition of `rdfs:subClassOf` derives new triple (`(:Kitty rdf:type :Animal)`). However, since this triple is based on triple (`(:Cat rdfs:subClassOf :Animal)`), the metadata to the derived triple are the same as those of triple (`(:Cat rdfs:subClassOf :Animal)`). Thus, triple (`(:Kitty rdf:type :Animal)`) is valid in version `:V1_1` and accessible only by user A.

3.2 Rules for Adding Metadata

We use the RDF toolkit, Jena [3] to implement this system and utilize the general purpose rule engine provided by Jena. As an example, the inference rule for `rdfs:subClassOf` is shown below, which shows that a new triple with the property of `rdf:type` is derived from the `rdfs:subClassOf` property.

```
[((?x rdfs:subClassOf ?y) (?a rdf:type ?x) -> (?a rdf:type ?y))]
```

Here, the syntax of Jena's rule engine is used to describe a rule². To handle the metadata, a new rule is defined to add not only a derived triple but also to attach an appropriate metadata to the derived triple.

The rule for adding metadata is defined to modify both the left and right hand sides of the original rule. Since there are many possible situations, let us consider a simple case where triple (`(:Kitty rdf:type :Cat)`) does not have metadata, but triple (`(:Cat rdfs:subClassOf :Animal)`) does.

The modified rule can be written as:

```
[((?x rdfs:subClassOf ?y) ((?a rdf:type ?x)
  (?_:b rdf:type Statement)(?_:b rdf:subject ?x)
  (?_:b rdf:subject rdfs:subClassOf) (?_:b rdf:object ?y))]
```

² <http://jena.sourceforge.net/inference/#rules>.

```
(?_:b sem:versionInfo ?version) (?_:b sem:accessInfo ?access)
noValue(?a rdf:type ?y) makeTemp(?r)
->
(?a rdf:type ?y)
(?r rdf:type rdf:Statement) (?r rdf:subject ?a)
(?r rdf:predicate rdf:type) (?r rdf:object ?y)
(?r sem:versionInfo ?version) (?r sem:accessInfo ?access) ]
```

The triple with the `rdfs:subClassOf` property is reified. The left hand side of the rule refers to the metadata, which is copied to the newly derived triple's metadata.

This rule can only handle a simple case where there are no metadata for the second triple (`:Kitty rdf:type :Cat`). In general, the metadata generated on the right hand side of the rule must be determined from the metadata of the triples on the rule's left hand side, which will be discussed below.

4 Version Management

The relationship between versions forms a tree-like structure, as shown in Fig. 3. When a new version is introduced, its ‘parent’ version is specified. Valid triples in the parent version are also valid in the child version. We assume that a version can have multiple parents.

The RDF resource that represents a version is an instance of `sem:version`. Each reified triple has metadata for version management as specified by `sem:versionInfo`, and its value is defined as an instance of `sem:version`.

The parent-child relationships between versions are represented by the `sem:parent` property. In addition, to facilitate judging whether a given triple is valid in a particular version, the version has an `sem:include` property that defines the relationship between versions. For a given version, the triples whose `sem:versionInfo` property value is included in the list of the `sem:include` properties of the version are considered valid in that version.

For the example in Fig. 3, the following relationships exist:

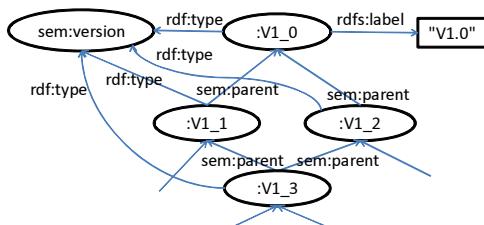


Fig. 3. Example version tree

```
:V1_0 sem:include :V1_0 .
:V1_1 sem:include :V1_1, :V1_0 .
:V1_2 sem:include :V1_2, :V1_0 .
:V1_3 sem:include :V1_3, :V1_2, :V1_1, :V1_0 .
```

This list shows that, for example, in version :V1_2, the triples in the parent version (such as :V1_0) are also valid. These `sem:include` relationships can be defined using the following rules. Note that the version ‘includes’ itself to simplify the query rewriting, which will be described later.

```
[(?c rdf:type sem:version) -> (?c sem:include ?c)]
[(?c sem:parentVersion ?p)(?p sem:include ?v)
 -> (?c sem:include ?v)]
```

In the example in the previous section, we assume that triple (`:Kitty rdf:type :Cat`) does not have metadata. We also assume here that the triple has metadata that indicate that the triple belongs to version (`:V1_2`). Since triple (`:Cat rdfs:subClassOf :Animal`) holds in `:V1_1`, derived triple (`:Kitty rdf:type :Animal`) should have the version that is the union of `:V1_1` and `:V1_2`. In the example of the version tree (Fig. 3), that version corresponds to version `:V1_3`. Thus, triple (`:Kitty rdf:type :Animal`) will have the version metadata for `:V1_3`. In the implementation, a specific rule is defined to handle this case.

Let us consider the following query in SPARQL:

```
SELECT DISTINCT ?type WHERE { :Kitty rdf:type ?type . }
```

When we make a query regarding version `:V1_3`, the query will be rewritten as follows:

```
SELECT DISTINCT ?type WHERE {
  :Kitty rdf:type ?type .
  ?r rdf:type rdf:Statement; rdf:subject :Kitty;
    rdf:predicate rdf:type; rdf:object ?type;
    sem:versionInfo ?version .
  :V1_3 sem:include ?version . }
```

The last clause checks the version. The result of the rewritten query is `:Cat` and `:Animal`. When we make a query regarding a different version, say `:V1_1`, the last clause will be changed to use `:V1_1` instead of `:V1_3`. In this way, the query can be rewritten to make a query only about the specified version.

5 Access Control

In addition to version management, the metadata attached to the reified triple are used to handle access control, whose metadata are specified as the property of `sem:accessInfo`.

5.1 Metadata for Access Control

Access permission to a triple is specified as metadata when the triple is added. The permission is represented using `sem:accessInfo` property. It has a property of `sem:permission` to denote the range of people who can access the given triple. We currently consider the following options: `sem:private`, `sem:public`, and `sem:group`. `sem:private` denotes a triple that only the owner (creator) can access, and `sem:public` denotes that a triple can be accessed by anyone. `sem:group` means that the triple can be accessed by a member who belongs to the same group as the owner. To determine if a user belongs to the same group, the user's profile is used. The user profile, which is represented in FOAF, must contain the information regarding the group to which the user belongs.

5.2 Query Rewriting for Access Control

Let us consider how to rewrite a query taking access control into consideration. Using the example in the previous section, we assume user B (different from the user who defined the triples in the example) makes a query to the RDF database. To simplify the explanation, we do not consider version management here.

The original query, given as

```
SELECT DISTINCT ?type WHERE { :Kitty rdf:type ?type . }
```

can be rewritten as follows:

```
SELECT DISTINCT ?type WHERE {
  :Kitty rdf:type ?type .
  ?r rdf:type rdf:Statement; rdf:subject :Kitty;
    rdf:predicate rdf:type; rdf:object ?type;
    sem:accessInfo ?access;
  ?access sem:ownedBy ?owner .
  { { ?access sem:permission sem:public . } UNION
    { ?owner foaf:name "B" . } UNION
    { ?access sem:permission sem:group .
      ?group foaf:member ?owner .
      ?group foaf:member ?member .
      ?member foaf:name "B" . } } }
```

In the rewritten query, the access condition is checked. If its permission is `sem:public` or the owner is user B, the triple can be accessed. Alternatively, if the owner and user B belong to the same group, the triple can be accessed.

In the example in Fig. 2, if user B makes a query, triple (`:Cat rdf:type :Animal`) cannot be accessed. Thus, the result of the query is `:Cat`. If user A makes a query, triple (`:Cat rdf:type :Animal`) can be accessed, and the result is `:Cat` and `:Animal`.

5.3 Updating Metadata

As in the case for version management, `sem:accessInfo` needs to be propagated when a new triple is derived by the ontology rules. Following the previous example, suppose that triple (`:Kitty rdf:type :Cat`) contains metadata for access control and that they specify an owner as user B with the permission of `sem:private`. In this case, triple (`:Kitty rdf:type :Animal`) cannot be derived because the triples that match the left side hand of the rule are created by different users with `sem:private` permission. In this case, only the metadata for `sem:accessInfo` are created and there are no `sem:permission` data. As another example, if user B specifies the range of access as `sem:group` instead of `sem:private` and user B belongs to the same group as user A, then the metadata for the newly derived triple can be accessed by user A, since it is assumed that users A and B belong to the same group.

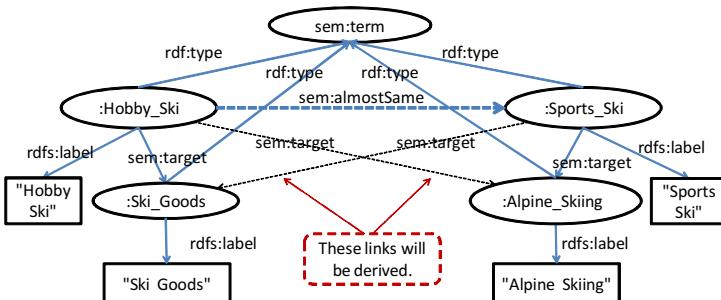


Fig. 4. Part of RDF topic database for “Easy-Free Conversation”

6 Application to Topic Database

The proposed mechanism is intended to be applied to the construction of a topic database that contains a word list used in conversations with people with communication handicaps such as aphasia. This topic database is converted from an original word list called ‘Easy-Free-Conversation’ [12] into an RDF database [9].

In this topic database, words are basically connected based on a conceptual hierarchy since the original word list was constructed in a tree hierarchy. During conversation, a word is selected and presented from the topic database following the links between the words. To facilitate conversation, links must be made between related words. If similar words are in the distant part of the tree hierarchy, it is difficult to follow them without direct links that connect them. Moreover, since the words often used in everyday conversations vary from person to person,

the topic database should contain some personal words. Thus, the topic database should be constructed by combining a database that has general words and a personal topic database (word list).

In an RDF topic database, a word is represented as an instance of a `sem:term`, and links are defined between them. The term has a `rdfs:label` property that contains its string representation. The property `sem:target` represents the relationship between terms to connect terms. Figure 4 shows a part of the topic database.

In this example, term `:Hobby_Ski`, which is under the category of hobbies, and term `:Sports_Ski`, which is under the category of sports, are shown. For each term, using the `sem:target` property, related words (`:Ski_Goods` and `:Alpine_Ski`) are connected. Here, ski under hobby (`:Hobby_Ski`) is not connected to the related words of ski under sports (`:Sports_Ski`) and vice versa, even though terms `:Hobby_Ski` and `:Sports_Ski` have similar concepts.

To define the relationship between these two terms, a new property `sem:almostSame` is introduced. The property's domain and range are `sem:term`. In this example, `:Hobby_Ski` and `:Sport_Ski` are linked by this property to reach more diverse terms so that we can handle a wider variety of conversation topics. The `sem:almostSame` property adds the `sem:target` links as expressed in the rule form below:

```
[ (?x sem:almostSame ?y) (?x rdf:type sem:term)
  (?y rdf:type sem:term)
  -> (?y sem:almostSame ?x) ]
[ (?x sem:almostSame ?y) (?x rdf:type sem:term)
  (?y rdf:type sem:term) (?x sem:target ?term)
  -> (?y sem:target ?term) ]
```

Here, the `sem:almostSame` property is defined to have a symmetry property (shown in the first rule). These rules do not consider the metadata. As in the examples shown in the previous sections, the rules are extended to handle metadata in the implementation.

Let us assume that user A adds the following statement in version `:V1_1` and grants access permission to the users who belong to the same group:

```
:Hobby_Ski sem:almostSame :Sports_Ski .
```

The query to obtain terms that can be accessed from the term `:Hobby_Ski` can be written as follows:

```
SELECT DISTINCT ?term WHERE { :Hobby_Ski sem:target ?term . }
```

When user A makes a query for version `:V1_1`, this query is rewritten to include the conditions for version management and access permission. The results are `:Alpine_Skiing` and `:Ski_Goods`. When user B, who belongs to the same group

as user A, the `sem:almostSame` property link becomes valid, and the same results can be obtained. However, for user C, who does not belong to the same group as user A, `sem:almostSame` property link does not hold, and only `:Ski_Goods` is obtained as a result.

In addition, if user A makes a query for version `:V1_0`, only `:Ski_Goods` is obtained as a result, since triple (`:Hobby_Ski sem:almostSame :Sports_Ski`) defined above is not valid in version `:V1_0`. In this way, a user can verify the effects of adding the property link of `sem:almostSame`.

7 Conclusion

This paper proposed a method for access control and version management for an RDF database that exploits the reification of RDF statements to add metadata and implements statement-level access control and version management. The application of the proposed method was described in the context of the construction of a topic database for conversation support where constructing a personalized database by combining several databases is useful. We are currently implementing the proposed method in an editing tool for the topic database, and we plan to evaluate the effectiveness of the proposed method in the development of the topic database.

References

1. Abel, F., De Coi, J., Henze, N., Koesling, A., Krause, D., Olmedilla, D.: Enabling advanced and context-dependent access control in RDF stores. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 1–14. Springer, Heidelberg (2007)
2. Au Yeung, C.M., Gibbins, N., Shadbolt, N.: Providing access control to online photo albums based on tags and linked data. In: Proceedings of Social Semantic Web, AAAI Spring Symposium 2009, pp. 9–14 (2009)
3. Carroll, J.J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A., Wilkinson, K.: Jena: implementing the semantic web recommendations. In: Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters, pp. 74–83 (2004)
4. Flouris, G., Fundulaki, I., Michou, M., Antoniou, G.: Controlling access to RDF graphs. In: Berre, A.J., Gómez-Pérez, A., Tutschku, K., Fensel, D. (eds.) FIS 2010. LNCS, vol. 6369, pp. 107–117. Springer, Heidelberg (2010)
5. Gergatsoulis, M., Lilis, P.: Multidimensional RDF. In: Chung, S. (ed.) OTM 2005. LNCS, vol. 3761, pp. 1188–1205. Springer, Heidelberg (2005)
6. Hollenbach, J., Presbrey, J., Berners-Lee, T.: Using RDF metadata to enable access control on the social semantic web. In: Proceedings of the Workshop on Collaborative Construction, Management and Linking of Structured Knowledge (2009)
7. Jain, A., Farkas, C.: Secure resource description framework: an access control model. In: Proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies, pp. 121–129 (2006)

8. Kim, J., Jung, K., Park, S.: An introduction to authorization conflict problem in RDF access control. In: Lovrek, I., Howlett, R.J., Jain, L.C. (eds.) KES 2008, Part II. LNCS (LNAI), vol. 5178, pp. 583–592. Springer, Heidelberg (2008)
9. Kuwabara, K., Shimode, Y., Miyamoto, S.: Agent-based remote conversation support for people with aphasia. In: Jędrzejowicz, P., Nguyen, N.T., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2010. LNCS, vol. 6070, pp. 371–380. Springer, Heidelberg (2010)
10. Ognyanov, D., Kiryakov, A.: Tracking changes in RDF(s) repositories. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) EKAW 2002. LNCS (LNAI), vol. 2473, pp. 373–378. Springer, Heidelberg (2002)
11. Reddivari, P., Finin, T., Joshi, A.: Policy based access control for a RDF store. In: Proceedings of the Policy Management for the Web Workshop (2005)
12. Yasuda, K., Nemoto, T., Takenaka, K., Mitachi, M., Kuwabara, K.: Effectiveness of a vocabulary data file, encyclopaedia, and internet homepages in a conversation-support system for people with moderate-to-severe aphasia. *Aphasiology* 21(9), 867–882 (2007)

Knowledge Elicitation Methods Taxonomy: Russian View

Tatiana A. Gavrilova, Irina A. Leshcheva, and Maria N. Rumyantseva

Graduate School of Management, St. Petersburg University, Volkhovsky per., 3,
199004 Saint-Petersburg, Russia

{gavrilova, leshcheva}@gsom.pu.ru, mrumyantseva@mail.ru

Abstract. The paper presents general classification of knowledge elicitation methods first and continues with elaboration of each individual method. In this paper we attempt to blend together Russian knowledge elicitation methods and practices with internationally applied methods. We concentrate on the analysis of knowledge elicitation methods of qualitative research. The result is a more comprehensive classification of knowledge elicitation methods. The classification further benefits from the authors' rich practical experience of implementing these methods.

Keywords: Knowledge management, knowledge engineering, classification of knowledge elicitation methods, practical application.

1 Introduction

Contemporary qualitative research methods are commonly perceived as developed in the Western society. They have indeed received broad practical application and became fashionable both in Europe and the United States. This led to a proliferation of publications and an army of researchers working to further refine qualitative methods. This, however, has not happened until the 1970s.

In a polarised world of the 20th century, qualitative methods were also actively developed behind the iron curtain. While Russian science has been traditionally perceived as advanced in highly quantitative fields, such as physics, mathematics, or applied statistics used for construction and control of five year plans, social sciences have been similarly advancing.

While a lengthier discourse into the social science development is certainly a very rewarding theme, this is not the focus on this paper. Here, we concentrate on a narrower topic of knowledge elicitation methods used in qualitative research.

2 Classification of Knowledge Elicitation Methods

A process of knowledge elicitation involves interaction between analyst extracting knowledge and expert possessing it. Although this is a common form of knowledge elicitation, it is not the only one available. Various authors [1-5] suggest over 30 verbal and computer-based methods of knowledge acquisition and processing. Such an abundance of methods seems to be somewhat excessive. In this paper we suggest a classification of commonly used methods of knowledge elicitation sorted by the source of knowledge they are dependent on (Fig. 1).

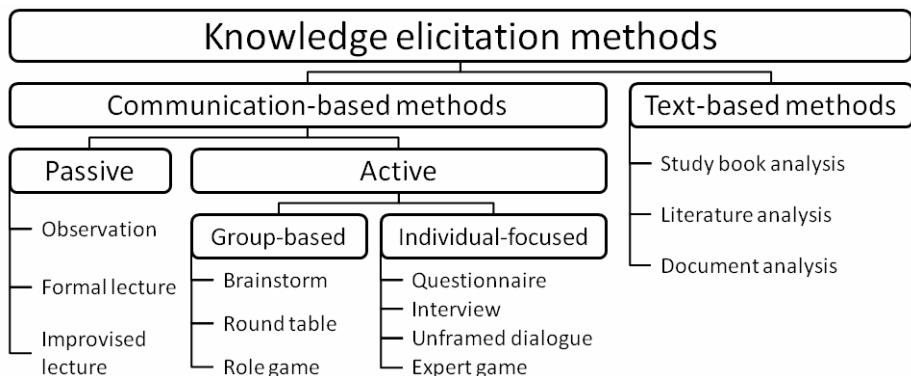


Fig. 1. Classification of knowledge elicitation methods and corresponding knowledge processes

Generally, communication-based methods of knowledge elicitation explain various interactions with one or several experts possessing knowledge. Text-based methods provide procedures for knowledge elicitation from documents and professional publications. The distinction between the two does not imply their incompatibility — these methods are frequently used in combination, where literature review is followed by expert interviews or vice versa.

Communication-based methods can be further divided into active and passive. Passive methods imply that the leading role in the knowledge extraction is passed on to the expert, where the role of the analyst is limited to observation and recording of expert's explanations. Alternatively, the decision-making process demonstrated by the expert could be presented in a form of a lecture. Contrary to passive, active communication methods imply that the person acquiring knowledge is in a full control of the process, actively communicating with the expert through various methods — games, dialogues, round table discussions, etc.

Both active and passive methods could be subsequently applied in one knowledge elicitation project. In a situation, where the person acquiring knowledge is less experienced in the field of the expert, he could first use passive methods, gradually applying more active techniques as he familiarises himself with the field.

Passive methods could be mistakenly regarded as more simple they, however, require specific skills to become effective. This is predominantly the ability of the analyst to process the stream of information and rightly detect valuable pieces of knowledge. The absence of a feedback loop significantly weakens passive methods, placing them as supplementary to active.

Active methods in their term could be divided into two groups, depending on the number of experts contributing their knowledge. If there is more than one expert, it is reasonable to combine individual contacts with group-based interactions. The latter are instrumental in stimulating knowledge processes and triggering new and nontrivial ideas. Nevertheless, individual-focused methods retain their dominance as the process of knowledge sharing is traditionally seen to be more efficient on a face-to-face basis.

3 Communication-Based Methods

In line with the classification of the Figure 1, both passive and active methods of communication are analysed as follows.

3.1 Passive Communication Methods

The notion passive is used here to stress different nature of these methods compared to active methods. Passive does not mean that these methods are less labour intensive, on the contrary, they require substantial effort comparable to such in active methods (e.g. organising games). Passive methods imply that the dominant role in the knowledge extraction is delegated to the expert, where the role of the analyst is limited to recording narratives of the expert throughout the decision making process.

Observation

The observation method implies that the analyst is located in the direct proximity to the expert, observing closely his professional activities or their imitation. Before a session is started the analyst should explain to the expert the purpose of the observation and ensure that the expert is commenting every decision he is making.

The analyst should record actions made by the expert, his comments and explanations. A video recording is usually helpful given that the expert has provided consent. The key precondition of this method is the avoidance of any intrusion by the analyst into the work of the expert. Due to this condition, the method is seen as the only “pure” method excluding interventions on the cognitive process by the observer.

The observation method is applied for both real processes of decision making and their imitations. In fact, both processes are frequently used together to allow for a higher level of detail. At first, the analyst observes a real-life process, deepening his understanding of the knowledge area and memorising visual attributes of the decision making procedure used by the expert. This follows by the observation of the process imitation, which is again performed by the expert at his work place, however this time the entire sequence of actions is performed solely for the sake of demonstration. The advantage of the imitation process is in that the expert is less stressed compared to the real process, where he simultaneously has to perform his job and demonstrate it to the analyst. However, this advantage is of a mixed nature — precisely the relaxed state of the expert can influence his results — as the work is only a simulation, decisions could vary from the real ones.

The scripts from the observation sessions should be transcribed in detail soon after the session and verified with the expert. In general, observation is one of the most common methods of knowledge elicitation on the earlier stages of this process. Its application is usually combined with the use of other methods.

Improvised lecture

The record of an improvised lecture is different from observation method in that the expert is asked to both comment on his actions and decisions and explain how

decisions were made, demonstrating logic used in decision making. While expert speaks, his “stream of thought” is carefully recorded, including pauses and exclamations. The method is sometime also called verbal reporting [6]. Whether recording devices should be used for this method is debatable as the influence of recording on the expert could be negative. If this is the case, it destroys a personal touch and an atmosphere of trust created in a face-to-face communication.

Transcript of protocols should be made by the analyst, observing the session. He is also the one correcting and refining the transcript after consequent sessions of knowledge articulation. A successful protocol of an improvised lecture is one of the most efficient methods of knowledge extraction, as it allows the expert to freely reveal his thinking process. He is free to show his erudition, demonstrate depth of his knowledge. Majority of experts considers this method as the most and pleasant.

It is fairly common that the improvised lecturing method is combined with one of the active methods to enable a feedback loop. This allows the interpretation created by the analyst to be verified by the expert.

Formal lecture

Lecturing is probably the oldest way of knowledge transfer. The art of lecturing has been highly regarded in science and culture since ancient times. However, our focus is not on the abilities to prepare and conduct lectures but rather abilities to listen, transcribe and understand it. As experts are usually available with or without experience and talent to lecture, the analyst has little influence over the choice of this method. If the expert has lecturing experience (e.g. he is a professor practicing in hospital or experienced manager of production facilities), the knowledge transfer in a form of a lecture could represent a concentrated and structured knowledge fragment.

Lecturing also allows significant degree of freedom, however, the topic and objectives of the lecture should be clearly formulated in advance. For instance, if the topic of a series of lectures is “Diagnosis of the flue”, the topic of a particular lecture is “symptoms analysis”, the objective — to teach audience how to diagnose the flue and forecast its development by using a set of characteristics listed by the expert. When the topic and objectives are provided, an experienced lecturer could structure his knowledge and refine the logic in advance.

The analyst will need to diligently transcribe the lecture and ask clarifying questions. High quality transcript includes tracking main points, omitting superficial details, good structuring of paragraphs and subparagraphs, care in articulation of clear and logical sentences, ability to generalise.

Good questions could also benefit both audience and lecturer. Thought-through questions could help gaining respect of the expert for the analyst.

As most passive methods, lecturing is an efficient method to quickly immerse the analyst into the area studied.

Comparative analysis of passive methods of knowledge extraction is provided below (Table 1).

Table 1. Comparative analysis of passive knowledge extraction methods

Observation	Improvised lecture	Formal lecture
Advantages		
Absence of analyst's influence. Maximum immersion of analyst into the knowledge area	Freedom of expression for the knowledge expert. Logic of the expert is well exposed. Absence of analyst's influence	Freedom of expression for the knowledge expert. Structured narration. High level of knowledge concentration. Absence of analyst's influence
Disadvantages		
Absence of a feedback loop. Knowledge fragmentation	Absence of a feedback loop. Possibility of a diversion of expert's narration from the desirable area of knowledge	Abundance of details. Weak feedback loop. Good lecturers are rare

3.2 Active Individual-Focused Communication Methods

Active methods of knowledge extraction are most frequently used. They are employed in development of most knowledge areas and imply active participation of the analyst, who writes scenario and facilitates knowledge extraction sessions. While expert games differ substantially from other methods in this group, the remaining three methods exercise a high degree of similarity. They are also classified as question-based.

Questionnaire

Questionnaire is a highly formalized method. The analyst formulates a list of questions in advance and provides them to a number of experts. The procedure could be organised in two ways.

- The analyst asks questions and records expert's answers.
- Following an instruction session, the expert completes questionnaire independently.

The choice depends on specific circumstances (e.g. questionnaire layout, clarity, readiness of the expert to spend time). The second method is usually favoured as it allows expert to spend unlimited time answering questions.

The analysis of communication process allows distinguishing between its three components: participants, means of communication and situation of communication. The usage of questionnaire allows influencing two components - the situation and the means of communication (i.e. questionnaire). There are several general recommendations for composing a questionnaire. An impressive experience of questionnaire application is accumulated in sociology and psychology [7, 8]. In the recent time, web-based methods of organising questionnaires are becoming increasingly popular. The electronic tools help to simplify data gathering and processing but still demand very careful preparation and financial investment.

Interview

Interview is a specific form of communication between the analyst and the expert, where analyst is asking a number of prepared in advance questions in order to gain better understanding of a specific knowledge area. Perhaps, the most comprehensive experience of interviewing is accumulated in journalism and sociology [9, 10].

The method of interviewing is close to the questionnaire method where the analyst records answers of the expert. The main distinction of the interviewing is in that it allows omitting some questions, adding new questions, changing pace of the interview, and generally enriching the content of communication. The analyst is able to employ non-verbal communication, use his charisma and better engage the expert in the interviewing process, therefore, improving the overall quality of the interview.

Unframed dialogue

Unframed dialogue implies a discussion between expert and analyst, which does not follow any specific plan or questionnaire. This, however, does not free the analyst from a profound preparation. On the contrary, a free-style and easy-going dialogue requires exceptional professional and psychological preparation. While preparation could vary in length depending on skills and experience of the analyst, it remains obligatory as it prevents from using the most irrational method of trial and error.

Profound preparation for the dialogue ensures that the analyst stays in control and skillfully directs communication. A well-planned discussion guarantees a smooth dialogue starting with a pleasant impression and a seamless transition to the main topics by installing interest and winning trust of the expert. The role of the analyst requires a very specific set of skills that have to be identified in the process of analyst's preparation. Choosing the right pace is central for the unframed dialogue.

Expert games

The game is the simulation of the professional activity. Expert games like any of the business games need the extraordinary amount of professional maturity from the analyst. The game design, scenario, preparation are really creative work. But the result may be outstanding as the game activates expert's mind and reveals his/her decision making procedures.

One of most simple expert game is imitation of a peer-to-peer dialogue between two specialists, where analyst is plays the role of a partner-specialist. The preparation in this case is similar to the preparation for an unframed dialogue.

Comparative analysis of active individual-focused methods of knowledge extraction is provided below (Table 2).

3.3 Active Group-Based Communication Methods

The group-based communication methods include role games, round table discussions and brainstorms. The main advantage of group-based methods is the simultaneous acquisition of knowledge from a number of experts. Their interaction increases the quality of the outcome by clashing different viewpoints and positions. As these methods are somewhat less frequently used than individual-focused methods (which is explained by the complexity of their organization) they will be explained in more detail.

Table 2. Comparative analysis of active individual-focused knowledge extraction methods

Questionnaire	Interview	Unframed dialogue
Advantages		
Uniformed survey of several experts. Minimal effort from analyst on the stage of data collection	Feedback loop (opportunity to clarify details and possible contradictions)	Flexibility. Strong feedback loop. Possibility to change scenario and form of communication
Disadvantages		
Experience and skills in questionnaire compilation. Absence of interface between expert and analyst. Absence of feedback loop. Questions could be misunderstood by the expert	Time consuming preparation of questions	Substantial stress for the analyst. Absence of formalized methodologies. Difficulty in recording the dialogue

Round table

The method of roundtable (the term is borrowed from journalism) prescribes a discussion of a given topic held by a number of experts, each given equal rights. Commonly, participants are asked to outline their positions first and move to an active discussion afterwards. The number of participant could vary from three to five. Most recommendations provided earlier in this paper are also applicable to this method. However, there is also some variation associated with human behaviour in a group.

The analyst is required to make additional effort of both organisational (e.g. preparation of location, coordination of time, place, quorum, refreshments, etc.) and psychological kind (e.g. ability to input relevant comments, sense of humour, good memory for names, ability to buffer conflicts, etc.).

The goal of the discussion is to analyse controversial hypothesis of the knowledge area collectively, from different viewpoints, under different research angles. In order to spice up the discussion, participants are invited from various scientific streams and generations. This limits the danger of obtaining one-sided knowledge.

Brainstorming

Active group-based methods are frequently used to revitalise the process of knowledge acquisition. On their own, these methods do not allow to source complete knowledge. They are usually applied as complementary to traditional individual-focused methods (e.g. observation, interview, etc.) to facilitate expert's thinking.

Brainstorming is commonly used to activate creative potential of participants. A brainstorming session does not last long (in the range of 40 minutes). Participants (up to 10) are invited to express their ideas of a given topic (no critique is allowed). An average number of ideas emerged in one session is about 50. Every participant has two minutes to express an idea. The most interesting part of the session is the point when the peak of the discussion is reached and ideas start to emerge at accelerated speed, leading to a simultaneous generation of hypotheses by many participants.

The analyst facilitating brainstorming session should be in full control of the audience, is responsible for a selection of active and creative experts, he should not

discriminate bad ideas as these could trigger thinking. A good facilitator is able to ask questions, stimulating idea generation process. Questions serve as a hook that helps to extract ideas [11]. They could also stop talkative experts from dominating discussion and facilitate development of ideas of the rest of the group. The main slogan of a brainstorming session is “the more ideas, the better”. Sessions are usually recorded.

Role games

Role games imply participation of several experts. The game is played according to a given scenario, all roles are assigned in advance, each role has a description and performance valuation matrix [12]. There are several methods of conducting role games. In some, participants decide what roles they want to play, in others they draw lots or receive assignments. A role is defined as a number of behavioral patterns, where each role is interlinked with the group behaviour within a game.

The number of roles is usually limited to three/six. If the number of experts exceeds the desirable group limit, they could be divided into two groups. Here, the competition between groups could further improve quality of the outcome. Enabling the right atmosphere for the game requires significant effort and creativity from the analyst. The game is considered a success if participants start identifying themselves with their roles.

4 Text-Based Methods

The group of text-based methods combines knowledge elicitation methods based on investigation of scientific texts, monographs, papers, and other written sources of professional knowledge. Among knowledge extraction methods, this group is least developed with little published instructions available. The following interpretation is, therefore, an introduction into these methods as it is seen by the authors.

When analysing a text, the analyst has to decompose it into categories in order to select knowledge fragments, important for his work. The rigour of text interpretation is also in understanding of contexts surrounding the text. There are micro and macro contexts. Micro context is the nearest surrounding of the text, such as paragraph, chapter, etc. Macro context is the entire knowledge universe associated with the particular knowledge field. In other words, the knowledge becomes meaningful in the context of a larger knowledge. So how the process of understanding is organised? One of the possible answers is given below. In general, this scheme could be applied in any learning process.

Main components of understanding a text are:

- Introducing initial hypothesis of the text’s meaning (foresight)
- Clarifying terminology
- Formulating general hypothesis of the text’s context (knowledge)
- Further elaborating meaning of terminology and interpreting text’s fragments using general hypothesis (deduction technique)
- Forming initial logic of the text by building intrinsic connections between key words and fragments, as well as by introducing abstract notions, generalising specific knowledge fragments

- Correcting general hypothesis according to the knowledge fragments sourced in the text (induction technique)
- Acceptance of general hypothesis.

The understanding of a text implies use of both deduction and induction techniques. Such a twofold approach allows seeing the text as a knowledge unit of special kind, with its main features being connectivity, wholeness, completeness. The key points of the process are forming the logic of the text by identifying key words and final connection of key words into a unified semantic structure.

5 Knowledge Elicitation Session: Case Study

The above described classification may be used as a practical guide for the knowledge engineer. Several years ago the authors participated in a project which was aimed at developing expert system for the forecast of a possible blow in mining industry. The expert was a well-known in Russian doctor of Science Prof. Buch (83 years old), who had done a lot of successful expertise in this field.

The Russian specifics of professional dialogues comprise several features:

- All the unframed dialogues are normally very informal
- There are very few experts ready to share knowledge
- Expert has no honorarium for the interview
- Group-based methods are nor very popular

Taking into consideration all this factors knowledge engineers have to combine a bunch of methods to reach the goal.

First step (3 weeks) was devoted to the *text-based methods* targeted at superficial comprehension of domain and its specifics. The bibliography was recommended by the expert. The book texts are usually written with a focus on specific audience. Therefore, if the text is not aimed at a general audience, one would require some preparation in order to understand such a text. In this case, the path to the knowledge lengthens by one more step. Therefore, although text-based methods are practiced as a preparation for active knowledge extraction, they on their own require solid preparation.

Then the expert was asked to provide several forecasts based on the detailed maps of mines from Siberian region using *improvised lecturing* (5 days). These lectures were followed by series of interviews and *unframed dialogues* (2-3 weeks).

All sessions were tape recorded and then notes and protocols were transcribed and worked-out. They were discussed with the expert and later the expert system was developed. It should be noted that the knowledge base was very subjective and this system was called “Buch’s forecast”.

6 Conclusion

The methods discussed in this paper form a comprehensive collection of tools for knowledge elicitation procedures. Many of these methods have been observed and

practiced by the authors for decades. Interestingly, some of them are more popular in Russia while others are more broadly practiced in Europe and the United States.

Among methods traditionally used in Russia are passive methods of observation and lecturing and active individual-focused methods of questionnaires and interviews. While active group-based methods have also been practiced in Russia, they have not been formalised as a research method. It is only in the last two decades when the active group-based methods have been consistently promoted and implemented in Russia. In general, there is no clash between Russian and Western qualitative research methods. They share common sources of origin and seem to be highly compatible. The differences between these methods are only discernible in details.

In this paper we have attempted to provide a comprehensive classification of knowledge elicitation methods. By definition such methods are focused on the processes of knowledge externalisation. This makes the analysis provided in this paper somewhat limited. While both tacit and explicit knowledge and related processes have been considered, the emphasis remained on explicit knowledge. This limitation should be addressed in future studies. The authors believe that practical application of the spectrum of methods outlined in this paper will lead to further extension and refinement of the classification.

References

1. Boose, G.H.: A survey of knowledge acquisition techniques and tools. *Knowledge Acquisition* 1, 3–37 (1989)
2. Gullen, J., Bryman, A.: The Knowledge Acquisition Bottleneck: Time For Reassessment. *Expert Systems* 5(3), 216–225 (1988)
3. Kendal, S., Creen, M.: An Introduction to Knowledge Engineering. Springer, US (2007)
4. Jones, S.R., Miles, J.C., Read, M.W.: A comparison of knowledge elicitation methods. *Expert Systems* 13(4), 277–295 (2000)
5. Cooke, N.J.: Varieties of Knowledge Elicitation Techniques. *International Journal of Human Computer Studies* 41(6), 801–849 (1994)
6. Morgoev, V.: Knowledge acquisition and structuring method: the consulting simulation. In: *Man-Machine Decision Support Systems*, pp. 44–57. VNIISI, Moscow (1988) (in Russian)
7. Anderson, M.L., Taylor, H.F.: *Sociology: The Essentials*. Cengage Publishers, NY (2010)
8. Yadov, V.A.: *Strategy of Sociology Study*, Moscow, Dobrosvet P.H. (2003) (in Russian)
9. Belanovsky, S.A.: *Individual Deep Interview*. MSU Press (2000) (in Russian)
10. Hashem, A.: *Interview Manual*. Ramesh Publishing House (2008)
11. Oppenheim, A.N.: *Creative Thinking and Brainstorming (Management skills library)*. J. Geoffrey Rawlinson Gower Publishing Ltd. (Paperback–3 April 1986) (1986)
12. Newstrom, J., Scannell, E.: *The Big Book of Business Games: Icebreakers, Creativity Exercises and Meeting Energizers*. McGraw-Hill, New York (1995)

A Formal Framework for Declarative Scene Description Transformation into Geometric Constraints

Georgios Bardis¹, Dimitrios Makris¹, Vassilios Golfinopoulos¹,
Georgios Miaoulis^{1,2}, and Dimitri Plemenos

¹ Department of Informatics, Technological Educational Institute of Athens
Ag. Spyridonos St., 122 10 Egaleo, Greece

² XLIM Laboratory, University of Limoges, 83 rue d'Isle,
Limoges, 87000, France

{gbardis, demak, golfinopoulos, gmiaoul}@teiath.gr

Abstract. The disambiguation of a declarative visual model is a crucial step towards the generation of its geometric equivalents. Any abstract description has to be ultimately translated into a concrete set of values or relevant constraints, solely relying on quantifiable model characteristics. To this day, there is no general consensus with respect to a unified, formally defined model for this disambiguation process or the desired quantified outcome. The current work sets the basis for a uniform transformation process and the corresponding formal constraint model, inclusive of a number of already existing approaches for declarative modelling. The applicability of the proposed framework is exhibited in an existing declarative modelling environment, explicitly demonstrating its implementation in the specific context.

Keywords: Declarative Modelling, Geometric Modelling, Constraints, Design.

1 Introduction

Declarative Scene Modelling [16] represents an early-phase design methodology where a scene is described using abstract and, typically, ambiguous terms instead of concrete geometric properties and values. It offers the advantages of a scene model based on terms closer to human intuition and the potential of unexpected innovative outcomes, its main drawback being the multitude of potential interpretations due to inherent model ambiguity and fuzziness. For the synthesis of the abstract model a number of alternative meta-models have been proposed, including but not limited to natural language, semantic networks and Horn clauses.

Despite the fact that the declarative description per se is, at times, considered a means towards understanding, processing or communicating the individual properties and characteristics of a scene, in the majority of relevant works its construction is only the first step towards the eventual synthesis of the corresponding visual result(s). Hence, typically, the initial description has to become subject of the appropriate operations, regarding its enhancement or transformation, before its computational processing and respective generation of its graphical counterparts. All relevant transformation and enhancement stages take place in an environment supporting the corresponding operations.

The formalisation of the transformations a declarative scene description has to undergo is presented in the current work, determining the conditions and implications inherent in its translation into a set of unambiguous constraints upon specific geometric properties. An explicit implementation of this formalisation is subsequently exhibited in the context of an operating declarative modelling environment.

2 State of the Art

Two major approaches exist to modelling mechanisms, namely *rule-based* and *imperative* scene modellers. In rule-based modellers the user's initial set of requirements is transformed with the aid of a set of rules [15] whereas in imperative modellers the user defines the construction of scenes through procedures following parameter values [10]. This is the case with the majority of commercial Computer-Aided-Design modellers, utilising a history-based model in order to represent complex scenes. The problem of both aforementioned approaches is the need, early during the scene designing process, for low-level details not relevant to the creative phase, mainly due to the lack of abstraction levels that would allow the user to validate ideas before resolving low-level issues. In declarative modelling, on the other hand, the initial abstract description is composed using intuitive terms provided by the scene design environment [16]. The geometric result(s) are automatically generated by mechanism(s) observing the constraints implied by the description.

The management and transformations of the initial description have various approaches in the context of declarative modellers. [2] proposes a meta-model incorporating a set of features, relations and specifications for the initial description serving as a semantic intermediate between the designer and the procedural level responsible for generating the corresponding virtual world. [18] presents an approach for terrain declarative modelling using five layers, each controlled by an initial intuitive user sketch, subsequently refined separately and automatically merged by a mechanism to produce a consistent detailed result. DE²MONS provides a general purpose modeller, where the user's description is translated in an internal model consisting of linear constraints [6]. PolyFormes is a specialised declarative modeller based on regular and semi-regular polyhedra where the user description is translated into an internal model which is expressed in terms of rules and facts, generating solutions through an inference engine applying rules to the facts, creating new facts and exploring the solution space [9]. MultiFormes [3], [16], [17] is based on declarative modelling by hierarchical decomposition (DMHD) [16] where the scene descriptions are input through dialog boxes allowing composition of a tree-like structure. A declarative design environment of NURB surfaces has also been proposed where an interfacing module is dedicated to translate one specific media into a structured semantic language by keywords extraction and knowledge organisation [4]. MultiCAD is an intelligent environment for scene modelling and design [11]. The description of a scene is transformed with the use of Enhanced Entity-Relational (E-ER) models combining the definition of a general enhanced ER model with certain notions of scene graphs.

3 Requirements Categorisation

Typically, the requirements connected with a declaratively described scene fall into one of two categories:

- **Strict:** This category includes requirements that are mandatory for all visual interpretations of the declarative description. Thus, the mechanism responsible for the generation of these interpretations, i.e. the *solutions*, has to be aware of these requirements and respect them during generation. This is generally achieved through a set of constraints reflecting these requirements.
- **Flexible:** This category comprises requirements that represent desired characteristics that are not critical for solution approval. The constraints reflecting these requirements usually suggest a *partial ordering* or *classification* of the solutions, according to performance against the specific constraint. The generation mechanism may or may not be aware of these, since they may be considered *observed qualities* of the produced solutions. The main differences and characteristics are summarised in Table 1.

Table 1. Requirements Categorisation

<i>Category</i>	<i>Scope</i>	<i>Result</i>	<i>Typical Stage(s)</i>
<i>Strict Requirements</i>	Mandatory	Validity	Generation
<i>Flexible Requirements</i>	Optional	Partial Order / Classification	Generation / Visualisation

The user-defined abstract description is often inadequate or extremely general for the generation of meaningful visualisations, since it is desirable to relieve the user from the incorporation of validating information and allow focusing on the user's functional, creative or aesthetic aspirations. As an example, the paradigm of Architectural design suggests that a declarative description of the user's desired habitation may not contain obvious facts (e.g. the requirement that rooms should not be isolated but connected with each other) that ensure the definition of a valid building assembly. These requirements are also declarative in their definition and can, therefore, be incorporated in the declarative description. Hence, the initial user description has to be enriched with elements that reflect the *domain knowledge*, additional features ensuring that the visual interpretation of the described scene will indeed represent a valid object. In addition, features of a specific subdomain may further enhance the description. For example, the description of a building supposed or expected to be in Naxos has to be enhanced with specific declarative elements reflecting the local architecture. Therefore, the description of an object in a specific knowledge domain may have to include elements of subdomain knowledge.

Moreover, one or more qualitative characteristics may also be of interest with respect to solutions generated and/or visualised according to given descriptions. These qualitative characteristics are typically connected with one or more measurable features of a solution for which the acceptable or preferred range of values may be dictated by user or other requirements. A summary of the aforementioned types of elements comprising an enhanced declarative description and example interpretations

are shown in Table 2. The task of generating alternative interpretations of a submitted description, possibly enhanced with additional requirements discussed in the previous section, requires *a set of constraints connected with measurable properties of the scene's visual interpretation(s)*. This set of constraints is a *transformed* version of the declarative description, appropriate for solution generation. Requirements from all types/categories of Table 2 have to be mapped to expressions explicitly containing measurable properties and operations among them. In the following we formalise these notions and demonstrate their implementation in a prototype declarative modelling environment.

Table 2. Requirement types and examples from Architecture

<i>Requirement Type\Category</i>	<i>Strict</i>	<i>Flexible</i>
<i>General Domain Knowledge</i>	Non-isolated spaces (Architecture)	
<i>Specific Subdomain Knowledge</i>	Linear arrangement of spaces (Traditional Architecture)	Oblong building (classification)
<i>Custom User Description</i>	Private/public zone separation (User)	Area (ranking)
<i>Other Aspects</i>		Building Compactness (ranking)

4 Description-to-Requirements Mapping Model

We hereby propose a declarative description model necessary for the connection with the corresponding constraints. This model, due to its construction, allows for the expression and concretisation of constraints from alternative domains. We start by examining a typical declarative description and its structural elements and then proceed to incorporate additional capabilities which are desired in order to support the entire range of requirements presented in the previous section. The typical declarative description D_D is a set of objects O_D , their properties P_D and associations among these objects A_D , i.e.

$$D_D = (O_D, P_D, A_D) \quad (1)$$

The initial declarative description is typically enhanced with additional features endowing it with the domain and subdomain knowledge necessary to restrict the outcomes within a desired locale, style, etc. Formally, three additional sets, O_K, P_K, A_K enhance the original description, thus leading to:

$$D = (O, P, A): O = O_D \cup O_K, P = P_D \cup P_K, A = A_D \cup A_K \quad (2)$$

Each of the aforementioned sets contains the entire set of objects, their properties and their associations of the enhanced description:

$$O = \{o_1, o_2, \dots, o_m\}, P = \{p_1, p_2, \dots, p_k\}, A = \{a_1, a_2, \dots, a_n\} \quad (3)$$

If $S(D)$ is the *maximal* set of geometric representations, each denoted by s , that fulfill D , then, by definition, it holds:

$$\begin{aligned}
& \forall s \in S(D) : \\
& s(o_1) \wedge s(o_2) \wedge \dots \wedge s(o_m) \\
& s(p_1) \wedge s(p_2) \wedge \dots \wedge s(p_k) \\
& s(a_1) \wedge s(a_2) \wedge \dots \wedge s(a_n)
\end{aligned} \tag{4}$$

where $s(o_i)$ signifies object o_i appears in solution s , $s(p_i)$ signifies declarative property p_i is fulfilled by solution s and $s(a_i)$ signifies declarative association a_i is fulfilled by solution s . In other words, all solutions in S comply with the *requirements* implied by D (in the subsequent section we elaborate on how this is achieved). This set of requirements comprises *strict* requirements implied by D . We use the notation

$$R(o_i), R(p_i), R(a_i) \tag{5}$$

to signify the requirements for presence of object o_i , fulfilment of property p_i and fulfilment of association a_i respectively. Based on the above, we denote

- $s(R)$ the fulfilment of requirement R by a solution s
- $R_O(D), R_P(D), R_A(D)$ the sets of requirements implied by the description's objects, properties and associations respectively
- $R_{ST}(D)$ the entire set of strict requirements posed by description D .

In addition to the strict requirements, a typical declarative description may be enriched with additional requirements that *rank* or simply *classify* the solution space and, thus, aid the improvement in quality of produced or visualised solutions. Hence, a declarative description D is typically connected with a set Q of such qualitative characteristics that can be observed on its visual representations.

$$Q = Q_c \cup Q_r \tag{6}$$

where Q_c are classifying characteristics, aimed to offer partial order(s) of the solution space, applicable to every solution by definition of Q and Q_r is the set of desired ranking characteristics. If

$$Q_r = \{q_{r_1}, q_{r_2}, \dots, q_{r_j}\} \tag{7}$$

we state that

$$\forall s \in S : s(q_{r_1}) \vee s(q_{r_2}) \vee \dots \vee s(q_{r_j}) \tag{8}$$

where $s(q_{r_i})$ signifies the fact that the ranking qualitative characteristic q_{r_i} is defined for solution s . We use $R(q_{r_i})$ to signify requirement for fulfilment of restrictive qualitative characteristic q_{r_i} and $R(q_{c_i})$ to signify requirement for observation and classification by qualitative characteristic q_{c_i} . We denote the fact that solution s fulfils flexible requirement R by $s(R)$ and $R_{FL}(D)$ the entire set of flexible requirements implied by D . Hence, a declarative description is mapped to a set of requirements:

$$\begin{aligned} D \rightarrow R_{ST}(D) \cup R_{FL}(D) \\ \forall s \in S(D), \forall R_i \in R_{ST}(D) : s(R_i) \\ \forall s \in S(D), \exists R_j \in R_{FL}(D) : s(R_j) \end{aligned} \quad (9)$$

Similarly, based on the contents of the enhanced description and the corresponding requirements, the following mapping also holds:

$$D \rightarrow R_o(D) \cup R_p(D) \cup R_A(D) \cup R_Q(D) \quad (10)$$

5 Requirements-to-Constraints Mapping Model

To connect the aforementioned requirements implied by D with its solutions $S(D)$, we use the set of *metrics* $M(S)$ containing *all* uniquely measurable characteristics of the solutions in $S(D)$. The domain of values for each metric with respect to a description D is the set containing all expected values for all possible solutions in $S(D)$. Without loss of generality, we assume the expected range for every metric to be the set \mathfrak{R} of real numbers, considering discrete values or subsets of \mathfrak{R} as special cases. Hence,

$$s \rightarrow \bar{v}(s) \text{ where } \bar{v}(s) = (v_1, v_2, \dots, v_n) \quad (11)$$

having $n=|M(S)|$ and $\bar{v}(s)$ containing, as dimensions, the values of each metric in $M(S)$ for the specific solution s , i.e. each solution is mapped to a unique vector $\bar{v}(s)$. We are now ready to define the transformation of any declarative description to a set of constraints that can be computationally processed towards the generation of the corresponding graphical representations. Formally:

$$R \rightarrow (f_R, c_R) \text{ where } f_R : V \mapsto I \quad (12)$$

i.e. each requirement is mapped to an ordered pair with $V \subseteq \mathfrak{R}^k$, $k \leq |M(S)|$ and $I \subseteq \mathfrak{R}^t$, t an arbitrary but specific positive integer. Notice that $k \leq |M(S)|$ and usually $k \ll |M(S)|$ for any R , in the sense that a typical requirement will generally refer to only a small subset of the solution's metrics. c_R represents the desired restriction with respect to the alternative values of constraint function f_R . Thus, each requirement R is transformed into an ordered pair comprising a multivariable constraint function f_R which maps the space of solution vectors (of k dimensions) to an arbitrary number of custom *indicator* vectors (of t dimensions) and a respective set of restrictions c_R for these indicators. For example, binary classification has $t=1$, f_R yielding values within the minimal range $\{0,1\} \subset \mathfrak{R}^1$ and c_R contains the restriction $f_R(s)=1$ implying higher ranking for solutions fulfilling it. In case of a flexible requirement suggesting partial ordering, e.g. through a percentage where *less* is better, we would have $t=1$ and range $[0,1] \subset \mathfrak{R}^1$, whereas c_R would contain the ordering rule:

$$s_1 \underset{R}{\leq} s_2 \Leftrightarrow f_R(s_1) \geq f_R(s_2) \quad (13)$$

where the first clause signifies that *solution s_1 is of poorer or at most equal performance to s_2 , according to requirement R* , and the second clause signifies the quantified

expression of this partial ordering. Similarly to the case of metrics, we have assumed the maximal expected range for every indicator to be the entire set of real numbers. The ordered pair (f_{R_i}, c_{R_i}) for any requirement R is the tool that allows verification of requirement fulfilment by any solution or its partial ordering according to it. Hence, a declarative description D is ultimately mapped to a set of constraints:

$$D \rightarrow \{(f_{R_1}, c_{R_1}), (f_{R_2}, c_{R_2}), \dots, (f_{R_N}, c_{R_N})\} \quad (14)$$

6 Constraint Functions: Practical Considerations

There are two major cases for the constraint functions f that greatly influence the design and practical implementation of a declarative modelling environment. In the first case, the constraints suggested by all requirements R_i representing a declarative description, i.e. all pairs (f_{R_i}, c_{R_i}) of Eq.14 are well known by domain knowledge, literature or other explicit sources. This implies that the participating metrics, any formula(s) interconnecting them and the corresponding restrictions are well defined for all requirements related with the given declarative description. In such a case, it is relatively straightforward to create a computational mechanism able to (a) evaluate solutions against all requirements of the description, (b) generate solutions that fulfil all strict requirements and at least one flexible requirement and (c) partially order solutions according to ordering flexible requirements. A mechanism responsible for the generation and evaluation of solutions could implement CSP methodologies [13] (classical, dynamic, etc. taking advantage of known functions and restrictions), evolutionary techniques [14] (taking advantage of f 's as fitness functions) or any other approach for the full or partial exploration of the solution space.

In the second case, at least one f is unknown, not previously defined. In this case, we will have to assume that at least the minimal form of information regarding this constraint will be available: an adequate number of example and counter-example solutions (if R_i is a strict or restrictive flexible requirement) or solutions sorted according to R_i (if R_i is a flexible classifying requirement). Hence, a machine learning mechanism may be assigned the task to *learn* requirement R_i , thus becoming itself the mapping function f_{R_i} . In general, the examples will be submitted to the learning mechanism in their full vector form. However, it may be the case that the total number of metrics is prohibitive of such an approach and, hence, pre-processing or additional information may be required to isolate the principal metrics contributing to the solutions' performance against R_i .

7 Example Implementation

The aforementioned theoretical framework covers the needs of a fully functional declarative modelling environment [1],[5],[8],[11]. Example requirements of all categories are visualised in the following and expressed in a quantifiable manner, operating under the assumption that, for any given description D , we have

$$M(S) = \bigcup_{i=1}^n \{x_i, y_i, z_i, l_i, w_i, h_i\} , \quad n = |O| \quad (15)$$

i.e. the metrics used for the quantification are the coordinates of the origin of every object in D as well as its length, width and height. The initial stage concerns the enrichment of the user's description according to domain-specific knowledge. In this case, the coherent description of a building demands several architectural constraints. The constraints are interpreted as restrictions in a building description, [7]. (Fig. 1).

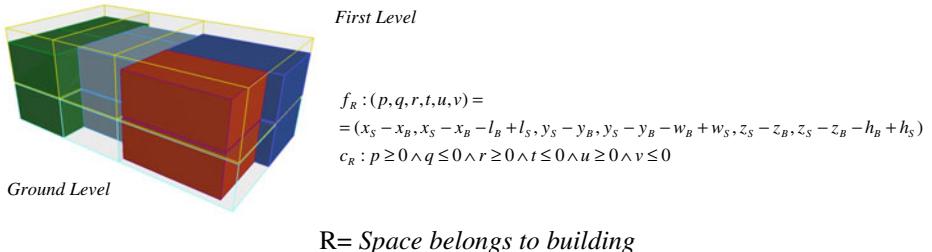


Fig. 1. Requirement R and corresponding constraint (f_R, c_R) for space S that must belong to a building B, suggested by domain knowledge

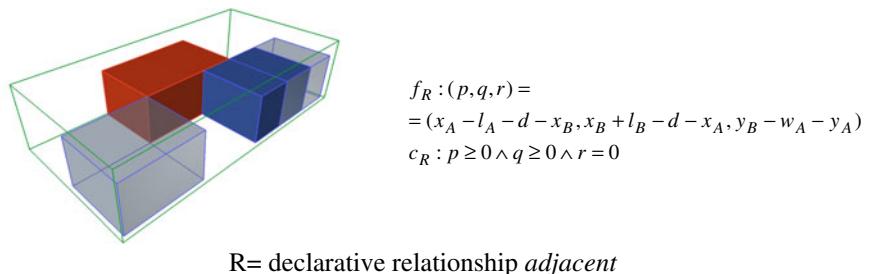


Fig. 2. Requirement R and corresponding constraint (f_R, c_R) for Kitchen (blue-A) adjacent to dining room (red-B) (d is a constant suggested by subdomain knowledge)

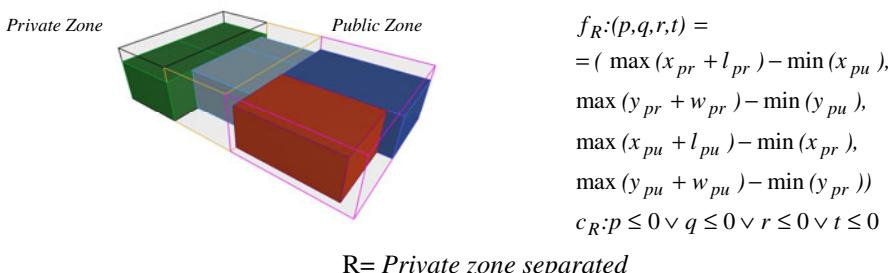


Fig. 3. Requirement R and corresponding constraint (f_R, c_R): the rooms of the private zone have to be placed separated from the rest of the zones, according to user demands

During this stage the output from the aforementioned stage may include additional requirements originating by a specific subdomain of interest. For example, we provide a requirement apparent in spatial arrangement of buildings in Naxos [12]. The requirement defines a declarative relationship, that “Kitchen must be adjacent to the Dinning room”, up to particular degree that it is defined by the constant d , (Fig. 2).

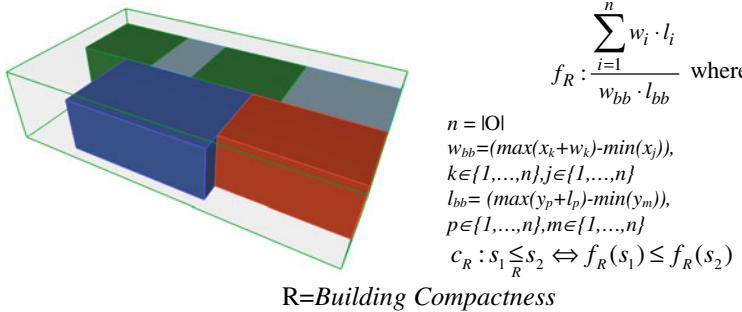


Fig. 4. Requirement R and corresponding constraint (f_R, c_R) for Building Compactness (higher is better), suggested by aesthetic aspects

The declarative description could bear particular user requests, e.g. “the rooms of the private zone to be placed separated from the other zones” (Fig. 3). The aesthetic appeal of a building also plays a crucial role in its final perception. A qualitative requirement might be that of “building compactness”, (Fig. 4) where the outer bounding box should leave minimal empty space outside the limits of the rooms.

8 Conclusions

The current work proposes an integral framework towards the universal formalisation of the declarative model disambiguation process through a unified constraint model, applicable to any declarative modelling environment. Despite the crucial role it plays in the visualisation process, disambiguation has not been adequately addressed in the relevant literature mainly due to its close connection with domain and subdomain knowledge as well as each user’s personal interpretations and preferences. Depending on the domain and user concepts, this kind of knowledge is often available only in empirical or ad hoc forms thus rendering the disambiguation process difficult to automate and formalise.

For this reason, we have defined the concrete set of transformations that have to take place and the minimal set of assumptions that have to be fulfilled in order to map a declarative scene description to a set of requirements. We have emphasised on the connection of these requirements with measurable result properties and have examined the role of the relevant existing knowledge in the construction of an appropriate solution generation mechanism. The applicability of the proposed framework has been demonstrated at several knowledge levels (domain, subdomain, user specific, other aspects) in an existing declarative modelling environment.

References

- [1] Bardis, G.: Intelligent Personalization in a Scene Modeling Environment. In: Miaoulis, G., Plemenos, D. (eds.) Intelligent Scene Modelling Information Systems. SCI, vol. 181. Springer, Heidelberg (2009) ISBN 978-3-540-92901-7
- [2] Bidarra, R., et al.: Integrating semantics and procedural generation: key enabling factors for declarative modeling of virtual worlds. In: Proceedings of the FOCUS K3D Conference on Semantic 3D Media and Content, France (February 2010)
- [3] Bonnefoi, P.-F., et al.: Declarative modeling in computer graphics: current results and future issues. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3039, pp. 80–89. Springer, Heidelberg (2004)
- [4] Estratat, M., et al.: An interfacing module using configuration for declarative design of NURBS surfaces. In: 3IA 2006, Limoges, pp. 85–96 (2006)
- [5] Golfinopoulos, V.: Understanding Scenes. In: Miaoulis, G., Plemenos, D. (eds.) Intelligent Scene Modelling Information Systems. SCI, vol. 181. Springer, Heidelberg (2009) ISBN 978-3-540-92901-7
- [6] Kwaiter, G.: A General Approach to Constraint Solving for Declarative Modeling Domain. In: International Conference on Information Visualisation, vol. iv, p. 424 (1999)
- [7] Makris, D., et al.: Towards a domain-specific knowledge intelligent information system for Computer-Aided Architectural Design. In: 3IA 2003, Limoges, pp. 65–78 (2003) ISBN 2-914256-06-X
- [8] Makris, D.: Aesthetic–Aided Intelligent 3D Scene Synthesis. In: Miaoulis, G., Plemenos, D. (eds.) Intelligent Scene Modelling Information Systems. SCI, vol. 181. Springer, Heidelberg (2009) ISBN 978-3-540-92901-7
- [9] Martin, D., et al.: PolyFormes: software for the declarative modeling of polyhedra. *The Visual Computer*, 55–76 (1999)
- [10] van der Meiden, H.A., et al.: Solving topological constraints for declarative families of objects. *Computer-Aided Design* 39(8), 652–662 (2007)
- [11] Miaoulis, G.: Contribution à l'étude des Systèmes d'Information Multimédia et Intelligent dédiés à la Conception Déclarative Assistée par l'Ordinateur – Le projet MultiCAD, Thèse de Doctorat, Université de Limoges, France (2002)
- [12] Michelis, P.: The Greek Traditional House. EMP Press, Athens (1981)
- [13] Miguel, I.: Dynamic Flexible Constraint Satisfaction and its Application to AI Planning. Springer Distinguished Dissertations Series (2004)
- [14] Mitchell, M.: An introduction to Genetic Algorithms. MIT Press, Cambridge (1998)
- [15] Muller, P., et al.: Procedural modeling of buildings. *ACM Transactions on Graphics* 25(3), 614–623 (2006)
- [16] Plemenos, D.: A contribution to study and development of scene modeling, generation and display techniques - The MultiFormes project. Professorial Dissertation, Nantes, France (1991)
- [17] Ruchaud, W., et al.: Multiformes: A declarative modeller as a 3D scene sketching tool. In: ICCVG, Zakopane, Poland (2002)
- [18] Smelik, R., et al.: Declarative Terrain Modeling for Military Training Games. *International Journal of Computer Games Technology* 2010 (2010)

User Movement Prediction Based on Traffic Topology for Value Added Services *

Marin Vukovic, Dragan Jevtic, and Ignac Lovrek

University of Zagreb, Faculty of Electrical Engineering and Computing,
Department of Telecommunications, Unska 3, HR-10000 Zagreb, Croatia
`{marin.vukovic, dragan.jevtic, ignac.lovrek}@fer.hr`

Abstract. Value added services are based on user context awareness. Important context aspect is location, which could be extended to future locations if services had the ability to predict movement. We propose a model for user movement prediction based on traffic topology. Benefits of the model are presented on example service, while the performance is evaluated on real user movement data.

Keywords: Movement prediction, traffic topology, value added services.

1 Introduction

Next generation telecommunication networks are turning to value added services in order to attract more users and increase revenues. Value added services use core telecommunication services, e.g. voice, data and messaging, for communication purposes but provide users with additional value, usually in form of user context related content and information. Important aspect of user context is location information while observing user locations over time results in awareness about user movement. Insight into user movement makes it possible to predict it as well, which opens up a whole array of new value added services and functionalities. The goal of this paper is to try and predict user movement in order to enhance value added services by extending the user context awareness to movement and movement prediction. In this sense, a model for movement prediction based on traffic topology is proposed.

In the scope of this work we examine outdoor user movement in urban and suburban areas. User movement is to some extent conditioned by the environment in which the user is moving. Environment parameters that condition user movement can be classified as obstacles and traffic related regularities. Obstacles can be natural, such as lakes, rivers, etc., or human made, such as fences, buildings or railway lines. Traffic related regularities are defined by traffic topology consisting of roads, railways, subway lines, bridges etc. Besides environment-conditioned movement, users also tend to move directionally, i.e. maintain their average movement direction. By gaining the

* This work was carried out within the research projects “Content Delivery and Mobility of Users and Services in New Generation Networks” and “Knowledge-based network and service management”, supported by the Ministry of Science, Education and Sports of the Republic of Croatia.

ability to estimate the user's direction and by combining such information with knowledge of the user's environment conditioned movement, it is possible to predict user locations and trajectories.

Next chapter gives an overview of related work regarding user movement prediction. Third chapter explains how traffic topology affects user movement and shows how it may be possible to predict user movement based on traffic topology and knowledge about user movement direction. User movement prediction model used for predicting locations and trajectories is proposed in fourth chapter. Since the proposed model targets value added services, chapter five gives an example of such a service and shows how it is possible to enhance services using movement prediction. Chapter six presents experimental evaluation carried out on real user movement data, while the last chapter concludes the work and discusses possible future work.

2 Related Work

The most common movement prediction application in mobile networks is prediction for reducing location management cost [1][2]. The goal is to predict where the users will move in order to reduce the need for frequent paging of user terminals thus reducing signaling traffic. Movement prediction is also used for various value added services [3][4]. Since these services rely on knowledge of user context to some extent, the possibility to expand such knowledge with user future locations gives them the ability to eventually adapt and deliver content with regards to user's future location.

Movement prediction in mobile networks implies the use of network and, recently, mobile terminal positioning techniques, such as GPS. Prediction for reducing location management costs usually relies on network positioning techniques that are less precise. On the other hand, applying prediction to location services may rely both on network and terminal based techniques that may result in more precise locations [5].

According to research in the field, user movement can be predicted by knowing user movement habits or just by observing user movement parameters at the time of the prediction, such as direction, speed etc. Prediction based on user knowledge is usually more complex because it involves analysis of user movement in order to detect regularities on which the prediction is based [4]. In [6] authors discuss and compare individual mobility patterns and show that individuals do indeed display significant movement regularity. Besides single user movement prediction, it may be possible to predict movement by analyzing group of users. Such analysis would result in regularities that are specific to the examined group and would enable movement prediction for the complete group at the time, although the prediction accuracy is expected to be considerably lower than single user based prediction. This is convenient for applications that need to predict movement for large number of users with lower requirements for prediction accuracy. However, both these concepts imply the existence of user movement data, so the user movement should be recorded for some period of time before the analysis and prediction is performed, which may raise privacy issues. On the other hand, movement prediction based solely on current user movement parameters has no such prerequisites, but also yields lower prediction accuracy.

Prediction model presented in [7] uses user speed, direction and distance in order to predict where the user will move, while the model presented in [8] tries to predict

the future location entry time as well. Interesting models are presented in [9] and [10] where authors base the prediction on road topology. The authors of both papers propose similar models that focus on bandwidth reservation in mobile networks. Mobile terminals report their precise locations to their current base station that holds a road topology map of the coverage area. Base station performs the prediction by using conditional probabilities of mobile terminals moving along predefined road segments within the station area. The goal of both of these models is to detect possible handovers to neighboring base stations in order to reserve or release bandwidth.

Movement prediction usually incorporates various techniques, mostly depending on the complexity of prediction and the existence of user movement data. In case of prediction based on user knowledge the techniques used for detecting regularities belong to the field of data mining [11]. Examined prediction techniques from the field use probability models and/or artificial neural networks. Probability methods usually build Markov models [5] and transition probability matrices [12].

An example of artificial neural network model is presented in [1] where authors distinguish three user categories based on whether it is possible to predict their movement. The prediction is done using multi layer perceptron (MLP) learned with backpropagation algorithm. Similar prediction concepts are presented in [13], where authors use two MLP networks; one for detection of regular movement and another for the prediction. On the other hand, authors in [12] used Bayesian learning for movement prediction and the model was evaluated on real user movement data from the Reality Mining project. Authors of [2] present a prediction model with the ability to group users based on their movement similarity, which is done using self organizing map (SOM), while the predictions are performed with MLP.

3 Traffic Topology and Movement

Traffic topology imposes various conditions on user movement, especially when user is moving along a street in a car, along a railway line in a train or similar. Obviously, pedestrians are the least conditioned by traffic topology, although it is expected that such users also move conditionally to some extent. In terms of movement prediction, we define roads, railway lines and similar traffic topology based paths as regular trajectories. Prerequisite for user movement prediction is the definition of such regular trajectories for geographical area in which the prediction is being done.

Trajectory extraction and definition depends on traffic topology within each location, which is relative to the location size. Single location can contain multiple traffic trajectories, depending on its size. Trajectory $Traj_i$, spanning through locations $\{Loc_1, \dots, Loc_m\}$, is defined as an array of consecutive transitions $Trans$ between locations:

$$Traj_i = \{Trans_1, \dots, Trans_n\}$$

where each transition is defined as a pair of source and destination locations, $Trans_i = (Loc_j, Loc_{j+1})$, and n is number of transitions defining movement along trajectory $Traj_i$ in one direction. That given, $n = m+1$, where m is the total number of locations along the trajectory $Traj_i$.

By analyzing all trajectories contained within a geographic area a trajectory set is formed containing definitions of all significant trajectories within the area. Once the definition of trajectory set is complete it is possible to estimate the probabilities of user moving from one location to another, according to trajectories. Probability that estimates the certainty of user moving from location Loc_i at the current moment in time t to other locations in moment $t+1$ is defined as follows:

$$Prob(Loc_i)_t = \{ (Loc_1, P(Loc_1)), \dots (Loc_k, P(Loc_k)) \}_{t+1}$$

where k indicates number of locations to which the user can move from location Loc_i , according to previously defined trajectories based on traffic topology.

Furthermore, using the assumption that users are moving directionally makes it possible to estimate the probabilities of transitions from one location to another with more certainty. Insight into user direction is gained by knowing past ($t-1$) and current (t) location of user, defined by transition $Trans_t$, where $Trans_t = \{Loc_{t-1}, Loc_t\}$. In general, the most probable transition from Loc_t , based solely on directional movement, is location Loc_{t+1} if locations Loc_{t-1} , Loc_t and Loc_{t+1} can be connected with a straight line, geographically observed. In scope of traffic topology directional movement is referred to as movement along the trajectory derived from traffic topology. Combining the knowledge about user direction with topology based transition probabilities makes it possible to estimate more precise transition probabilities from a pair of consecutive locations.

4 Movement Prediction Model

User movement prediction model is shown on figure 1. Location prediction is the basis of the model and is done by multi layer perceptron (MLP), artificial neural network architecture with supervised learning. Neural networks are characterized by two phases: learning, in which the network learns input – output pattern's relations, and application, in which the network applies the knowledge gained in the learning process on input patterns. The movement prediction problem comes down to determining the future location for a given pair or single previous location. Since input and output patterns for this purpose are defined during the trajectory and probability definition process, supervised learning is used for this purpose. With supervised learning, it is possible to monitor and adjust the average prediction error during and after the learning process. This is important because it enables an adaptation mechanism, so the prediction model can be adapted to possible changes in traffic topology, e.g. road works and similar. The other important advantage of neural networks over conventional methods is the network's generalization property. In scope of movement prediction, generalization enables that if a new pattern is placed as a network input, the network will try to predict the most suitable future location based on similarity of new pattern with specific pattern from the learning set. This property is necessary for scenarios where user is moving near or alongside the learned trajectory.

The most important step before the learning of the network is input and output pattern definition. The learning set patterns are defined as follows:

Input pattern: Loc_{t-1}, Loc_t

Output pattern: $P(Loc_1), \dots P(Loc_n)$

where pair (Loc_{t-1}, Loc_t) denotes previous and current location, or previous transition. If there is no record of previous user location, the prediction is done with no regard to directional movement.

As shown on figure 1, location prediction MLP input layer consists of $2 \cdot n$ neurons, where n is the number of locations used for prediction in the examined geographic area. Activation of each input neuron indicates that user was $(t-1)$ or is (t) at a location specified by neuron's ordinal in the input pattern. Output layer consists of n locations that correspond to locations in the input pattern. Output neuron activation indicates possible transitions, and the value of activation corresponds to the probability of a user moving to location specified by output layer neuron. Size of the hidden layer depends on complexity of total input-output pattern relations and is best determined experimentally. The learning was performed using the back propagation algorithm.

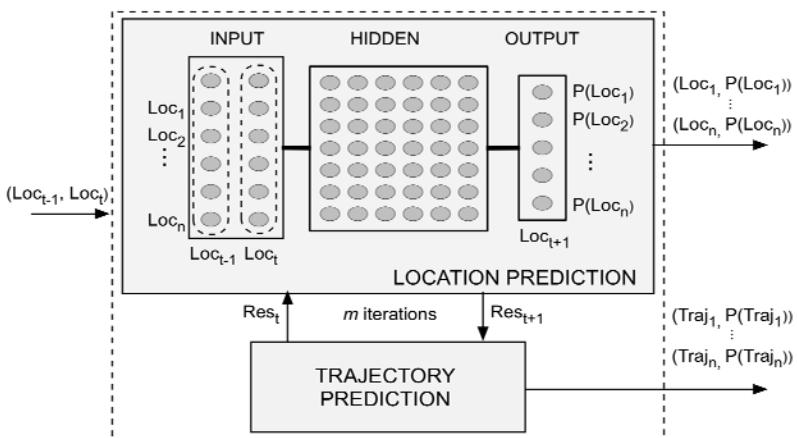


Fig. 1. Movement prediction model

After the learning process is complete, the network is able to predict user movement in form of next probable location(s). Input patterns that are brought to the network may identify single location or a pair of locations. Presentation of inputs as two consecutive locations yields higher prediction certainty because it uses information about user direction as well as traffic topology related trajectories.

Knowing the user's next location makes it possible to predict the user trajectory. This is done with proposed neural network model in such a way that the results of next location prediction in one iteration are used as inputs in next iteration. This requires prior definition of the desired number of locations on a trajectory. Once trajectory prediction is finished, the total probability of single trajectory is estimated by multiplying all transition probabilities contained within the trajectory.

5 Service Example

Service benefits from using movement prediction are presented on an example value added service. The service enables users to buy tickets for various events by using

their mobile phones. Users interact with service via short messages (SMS) in which they state their requests in form of completely free text. Some of the methods used for free text analysis are presented in [14]. The ticket is delivered to user's mobile phone via multimedia message (MMS) as 2-dimensional barcode. Basic service usage is shown on figure 2, for example of buying a cinema ticket. The main request parameters are the desired movie name, time, location and number of tickets. If any of these parameters are incorrect, missing or not applicable for the show, the service offers an alternative to the user thus engaging a dialogue.



Fig. 2. Basic usage of Paperless Ticketing service

While some parameters have to be explicitly defined by the user, such as movie name and number of tickets, some parameters can be suggested by the service if it is aware of the user context. If service has the ability to predict user movement, it may be able to suggest the location of the show. The service could try to examine user's current location, predict next location or even predict possible trajectories in order to find the most appropriate cinema location.

Table 1 depicts possible enhancements to the example service. In first scenario, service can predict where the user will be in next moment in time and suggest appropriate cinema location. Since the time of the show is not defined, the service concludes that the user wants tickets near to the time of the show. Second scenario represents a case where user enters time, movie name and number of tickets for the show. In this scenario, the service uses a trajectory prediction in order to find the most probable trajectory of the user in next hour, corresponding to time difference between the request and user desired time.

Table 1. Example service messages

User request and time	Service response (usual)	Service response (with movement prediction)
19:00 - "2 tickets for Star Wars"	"Please select location. Star Wars is showing..."	"You will buy 2 tickets for Star Wars at 19:15 in (predicted future location)..."
19:00 - "2 tickets for Star Wars at 20:00"	"Please select location. Star Wars are showing..."	"You will buy 2 tickets for Star Wars at 20:00 in (predicted location from trajectory)..."

6 Experimental Evaluation

Movement prediction model was evaluated using movement data of selected ten participants from the Reality Mining project [15]. The project was conducted at

Massachusetts Institute of Technology and its goal was to collect various user related data with an application installed on participant's mobile phones. In the scope of this work, the most interesting data collected is movement data recorded as consecutive mobile network cell identifiers.

The project participants were located in Boston, so the trajectories and probabilities should be based on Boston traffic topology. However, exact geographical positions of each cell recorded in the participant's movement data are not available. This was resolved by using labeled locations from each participant's movement record bound with transitions between such locations, extracted from participant's movement records. Participants had an option to give labels, i.e. names, to each location they visited so they usually labeled locations, i.e. cells, in which they spend significant amount of time (e.g. work, home, etc.) or other significant locations (e.g. subway stations, squares, etc.). Furthermore, participants often labeled several cells as single location, which is the consequence of handovers between neighboring cells. Combining trajectories between labeled locations, extracted from participant's movement records, and linking them to traffic topology results in a set of trajectories that are related to traffic topology and are applicable to all participants moving in the examined geographical area.

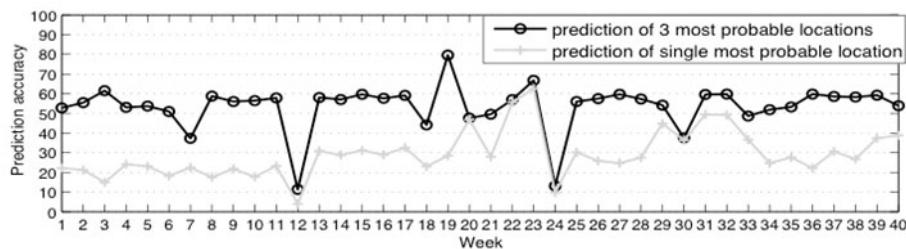


Fig. 3. Location prediction results for participant 1

Results of location prediction for participant 1 are presented on figure 3. Prediction results are shown per each week of user movement data, with values corresponding to average accuracy for the week. One line represents a case where the most probable predicted location is interpreted as a correct result. The other line represents a case when one out of three predicted locations was the actual next user location, because three neighboring cells, in average, often handover signal between each other, as observed from the recorded data. It is necessary to emphasize that interpretation of single, most probable, location as correct prediction result would always result in rather low prediction accuracy when mobile network cells represent locations. This is due to common handovers between neighboring cells in mobile networks, especially in urban areas where cell density is high. Grouping three possible cells as a positive outcome thus proves justifiable when evaluating prediction on data with high possibility of signal interference, i.e. handovers.

According to results on figure 3, the examined participant's movement is highly related to traffic topology included in the learning set. For example, 12th and 24th week show low accuracy, which leads to conclusion that the participant was not present in

the area included in the prediction. The average prediction accuracy over 40 weeks for participant 1 is 53%, when predicting three future locations.

Figure 4 shows average prediction accuracy for selected participants, which is around 30%, when predicting three most probable locations, or around 15%, when predicting one most probable location. Participants 1, 4, 7 and 10 show similar prediction results, which may point out a fact that they are moving along similar trajectories, given that the geographical area examined is rather small. During the project, participants filled in a questionnaire about their habits. These participants have all indicated they live near the faculty and go to work by foot or by subway, which supports the claim they actually move along similar trajectories. Furthermore, participant 8, with low prediction accuracy, indicated that he lives far from the faculty. This explains low accuracy, due to the fact that the area in which this participant moves was not taken into consideration during prediction.

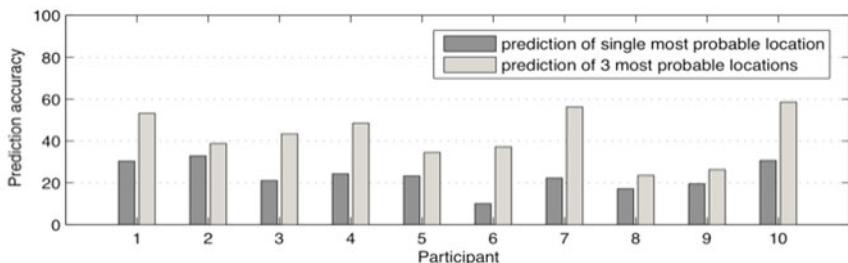


Fig. 4. Location prediction results for selected participants

Figure 5 shows trajectory prediction accuracy for participant 1. Presented values correspond to average accuracy per week, totaling 40 weeks. Since trajectory prediction is derived from location prediction, the presented results correspond to the results shown on figure 3. Obviously, the accuracy in this case is considerably lower, which is expected due to the iterative use of location prediction for this purpose. Very low accuracy is once again observed in 12th week. However, low accuracy from 22nd to 24th week does not fully correspond to location prediction results. This may be due to user actually moving through the locations covered by the prediction model, which is indicated by high accuracy for location prediction in week 23, but not following the usual trajectories, expected from most users. The average trajectory prediction accuracy for the complete period is 28%.

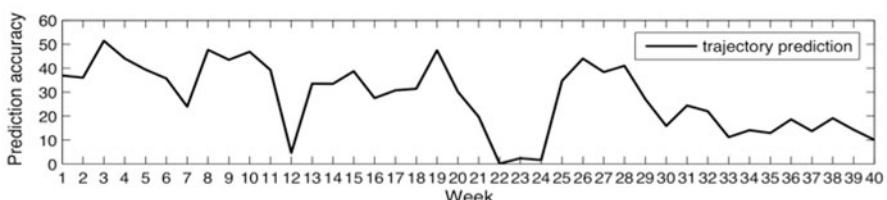


Fig. 5. Trajectory prediction results for participant 1

Figure 6 shows trajectory prediction results for selected participants. Once again, the results of trajectory prediction correlate with results of location prediction presented on figure 4, with worst results for participants 8 and 9. Prediction accuracy for all participants spans between minimum 5% and maximum 35 %, which is rather low. However, such results are expected because it is difficult to predict the precise trajectory of users without knowledge about the users themselves. Besides that, trajectories may consist of a large number of locations and the results shown here relate to users moving along complete trajectory, e.g. street, which may not be the case in the real world.

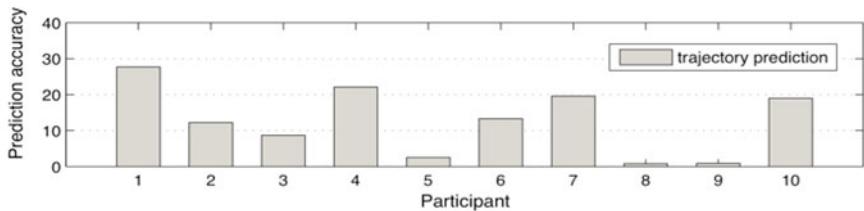


Fig. 6. Trajectory prediction results for selected participants

7 Conclusion and Future Work

The presented user movement prediction model is able to predict user locations and trajectories based on traffic topology. Possible benefits to value added services are presented on a service example and it is shown how such a model could enhance existing value added services. Experimental evaluation showed that it is indeed possible to predict real user movement with 30% success when predicting locations. It may be possible to increase the prediction accuracy but it would require a deeper insight into user movement habits, which may raise privacy concerns. In that sense, the advantage of presented model is that it does not require users to withhold their privacy any more than they already are by using wireless networks.

Regarding future work, it is necessary to emphasize that the presented prediction results were based on a segment of geographical area in which the participants were moving. In this sense, the prediction accuracy may be more precise if larger geographical area and its topology was taken into consideration. The plan is to examine this issue by tracking several users in a limited geographical area. Nevertheless, the presented results may be satisfactory for the purpose of value added services because these services need the information about user movement only in the vicinity of their service locations, e.g. in the city where the movies from example service are offered.

Another possible topic for future works relates to the process of acquiring traffic topology data, which may be complex for large areas. Thus, it is possible to select a group of users that are moving in the same area and anonymously track their movement. By overlapping movement from different users it may be possible to acquire regularities conditioned by traffic topology, and this topic will be addressed as future work as well.

References

1. Quintero, A.: A user pattern learning strategy for managing users' mobility in umts networks. *IEEE Transactions on Mobile Computing* 4(6), 552–566 (2005)
2. Majumdar, K., Das, N.: Mobile user tracking using a hybrid neural network. *Wirel. Netw.* 11(3), 275–284 (2005)
3. Krumm, J., Horvitz, E.: Predestination: Inferring Destinations from Partial Trajectories. In: Dourish, P., Friday, A. (eds.) *UbiComp 2006. LNCS*, vol. 4206, pp. 243–260. Springer, Heidelberg (2006)
4. Karimi, H.A., Liu, X.: A predictive location model for location-based services. In: *GIS 2003: Proceedings of the 11th ACM International Symposium on Advances in Geographic Information Systems*, pp. 126–133. ACM, New York (2003)
5. Ashbrook, D., Starner, T.: Learning significant locations and predicting user movement with gps. In: *Proceedings of the 6th International Symposium on Wearable Computers, ISWCi 2002* (2002) 0-7695-1816-8/02
6. González, M.C., Hidalgo, C.A., Barabási, A.-L.: Understanding individual human mobility patterns. *Nature* 453, 779–782 (2008)
7. Islam, M.M., Murshed, M., Dooley, L.S.: New mobility based call admission control with on-demand borrowing scheme for qos provisioning. In: *ITCC 2003: Proceedings of the International Conference on Information Technology: Computers and Communications*, p. 263 (2003)
8. Hou, J., Fang, Y.: Mobility-based call admission control schemes for wireless mobile networks. *Wireless Communications and Mobile Computing* 1(3), 269–282 (2001)
9. Hsueh, Y., Lee, D.: A bandwidth reservation scheme based on road information for the next generation cellular networks. *IEEE Trans. Veh. Technol.* 53(1), 243–252 (2004)
10. Kim, H., Soh, W.: Dynamic bandwidth reservation in cellular networks using road topology based mobility predictions. In: *Proceedings of the IEEE INFOCOM 2004*, vol. 4, pp. 2766–2777 (2004)
11. Yavas, G., Katsaros, D., Ulusoy, O., Manolopoulos, Y.: A data mining approach for location prediction in mobile environments. *Data Knowl. Eng.* 54(2), 121–146 (2005)
12. Liu, X., Karimi, H.A.: Location awareness through trajectory prediction. *Computers, Environment and Urban Systems* 30(6), 741–756 (2006)
13. Vukovic, M., Lovrek, I., Jevtic, D.: Predicting User Movement for Advanced Locationaware Services. In: *Proceedings of the 15th International Conference on Software, Telecommunications and Computer Networks, FESB*, University of Split. (2007)
14. Jevtic, D., Car, Z., Vukovic, M.: Location name extraction for user created digital content services. In: Apolloni, B., Howlett, R.J., Jain, L. (eds.) *KES 2007, Part I. LNCS (LNAI)*, vol. 4692, pp. 623–630. Springer, Heidelberg (2007)
15. Eagle, N., Pentland, A., Lazer, D.: Inferring Social Network Structure using Mobile Phone Data. *Proceedings of the National Academy of Sciences* 106(36), 15274–15278 (2009)

Emotion Judgment Method from a Meaning of an Utterance Sentence

Seiji Tsuchiya, Misako Imono, Eriko Yoshimura, and Hirokazu Watabe

Dept. of Intelligent Information Engineering and Sciences, Doshisha University,
Kyo-Tanabe, Kyoto, 610-0394, Japan
{stsuchiya, hwatabe}@mail.doshisha.ac.jp,
{eyoshimura, imono}@indy.doshisha.ac.jp

Abstract. Authors focus on the emotion of which common sense and attempt to compose a method that judge the user's emotions based on utterances. Such system and method have already been developed. However, emotions might not be able to judge from the existing method accurately because of the polysemy of the word. In addition, a lot of resources are needed to make the existing method and to maintain a knowledge base. Therefore, a method is not processing in meaning categories of words separately, but translating an utterance sentence into a word is proposed in this paper. Herewith, both a polysemy problem and the maintenance problem of the knowledge base are solved. The proposed method uses knowledge base and an Association Mechanism. As a result, the accuracy of the proposed method was improved approximately 18.4% compare with an existing method.

Keywords: Emotion, Common Sense, Concept Base, Degree of Association.

1 Introduction

Authors are conducting research aiming to develop new interfaces that follow the mechanism of human communication, focusing on human common sense. Humans, in such communication, are able to appropriately interpret ambiguous information that they receive and carry on a smooth conversation. Common sense is knowledge (ability) that only man has. The person can express feeling either sense of incompatibility or unnatural by using common sense. Moreover, the person can appropriately interpret when the sense of incompatibility and unnatural are felt.

Especially, authors focus on the emotion of which common sense and attempt to compose a method that judge the user's emotion, based on the utterance. For example, this system is able to output not an utterance which expressing sadness as "You are no good at anything" when a user say "Because I failed in work, it has been scolded by the superior" but a consoling utterance as "The odds are in your favor". Thus, using this system can select an appropriate expression when the content tries to provide the user contains expressions that are unpleasant or remind the user of unhappy events.

Such system and method have already been developed. The developed method [12] judges user's emotions, categorized into 10 types, from a sentence of the user utterance, based on the four components of the sentence: "subject", "modifier", "object word", and "action word".

However, emotions might not be able to judge from processing accurately because of the polysemy of the word. Thus, when meaning categories of object words and action words are judged separately, the polysemy cannot interpret. In addition, a lot of resources are needed to make the existing method and to maintain a knowledge base.

Therefore, in this paper, a method which dose not processing separately in meaning categories by object words and action words, but both object words and action words which used semantic feature are proposed. As a result, both a polysemy problem and the problem of the maintenance of the knowledge base can be solved. The proposed method uses knowledge base and an Association Mechanism.

2 The Existing Emotion Judgment System

The component parts of the utterance sentences are used to judge speaker's emotion were limited to four ("subjects", "modifiers", "objects" and "action words") [12]. Figure 1 shows outline of the existing Emotion Judgment method.

A "subject" is categorized into 3 attributes: liking (likes and dislikes), familiarity (closeness) and sociality (good and evil). These 3 attributes have 3 values. In short, "subject" is categorized into 27 categories. The categorization is processed using the knowledge base based on thesaurus [3].

A "modifier" is an adjective or "adjectival verb" that modify the "object" which follows the modifier. "Modifiers" is able to omitted, as they were not

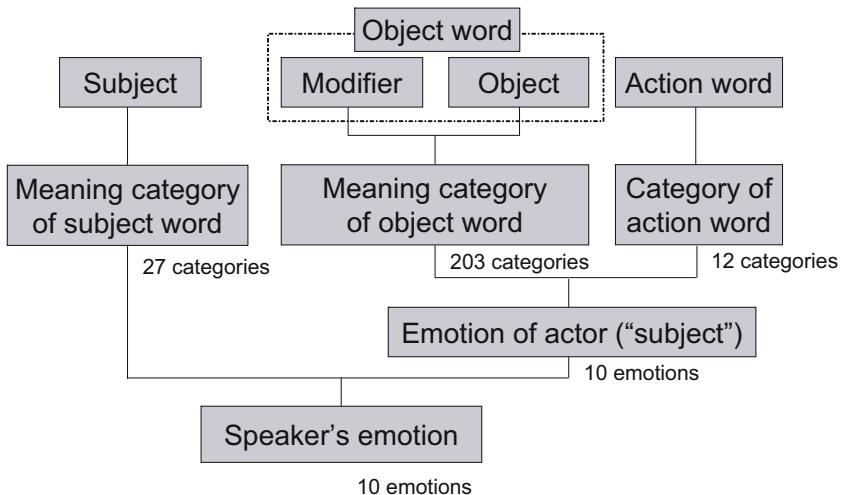


Fig. 1. Outline of the existing Emotion Judgment System

always necessary in textual expression. The direct modification and dependent modification types which divided into different groups having similar meaning, according to the adjectives describes the modifiers, and they were registered in the knowledge base for emotion judgment.

An "object" is a noun that denotes subject's action, behavior, or state. Objects were also classified according to their meanings using the 203 sense words, which the Sense Judgment System [4] can judge. These 203 sense words share the common meaning categories with the modifiers treat earlier. In addition, "modifiers" and "objects" refer collectively as "object words". In short, the 203 sense words are used for categorize the meanings of the object words.

Action words denote the speaker's action, behavior, and state. It include verbs, adjectives and adjectival verbs. For instance, the action word is in the sentence "My stomach aches", "aches". An action word converts the feature related to the sense and perception that associated with an object word. Features which expressed in terms of senses and perceptions are able to divided into positive and negative expressions roughly. For example, positive word expressions include "is beautiful", "is important", etc., and negative word include "ache" and "is dirty". Likewise, emotions can also categorize into two groups: positive and negative. For instance, "joy" and "ease" belong to the positive feelings, and "sadness" and "anger" belong to the negative feelings. Therefore, four types of effect can found in the action words (Figure 2).

The action words in [A] and [B] which shown in Figure 2 denote the unique emotions, and they doesn't depend on object words. Examples of words in [A] include "happy" and "enjoy", while those in [B] include "grieve" and "be afraid of". Words in [C] and [D] have different feelings depending on the meaning category of the object word. A word in the [C] category generates an emotion "succeeds" by the meaning category of the object word. In contrast, a word in the [D] category creates an emotion "opposite to" by the meaning category of the object word. Examples of words in [C] are "see" and "get", while the [D] group includes "lose" and "throw away". Hereafter, the action words in [A] and [B] will be refer to "unique-emotion" words and also those in [C] and [D] refer to "object word-dependent". The "unique-emotion" action words are further divided into

[A] Unique-emotion (positive)	[B] Unique-emotion (negative)
Object word Emotion + + - - + → + - → -	Object word Emotion + + - - + → + - → -
[C] Object word-dependent (succession)	[D] Object word-dependent (opposite)
Object word Emotion + + - - + → + - → -	Object word Emotion + + - - + → + - → -

Fig. 2. Concepts of the four types of action word interaction

10 sub-categories of emotions for the system to judge. For instance, words that indicate "happiness" are categorized in the "happy" sub-category and "sadness" are categorized in the "sad" sub-category.

This knowledge of action words was built with a thesaurus that systematically arranged verbs, according to the actions and states they represent. Thus, the knowledge base has registered all the self-sufficient verbs, as well as all 17,676 nouns indicating actions and states (those Japanese nouns that are frequently turned easily into verbs with a suffix). This thesaurus of verbs enables the system to classify large numbers of verbs with ease. In order to handle adjectives and adjectival verbs, the system uses its knowledge of modifiers.

A sentential actor's ("subject") emotion was judged based on the "object words", and "action words". With respect to the emotions that were generated, those associated with a total of 406 pairs of the meaning categories of object words (203 categories) and action words (2 categories of "succession" and "opposite"). They were manually defined and registered in the knowledge base. Therefore, speaker's emotion is judged by combination of attributes values (27 categories) of sentential actor ("subject"), and judged emotions of sentential actor. 270 rules are combination of 27 categories (attributes values) of sentential actor and 10 emotions are registered in the knowledge base.

Some knowledge related to the "generation of emotion", the "action words", and the "modifiers" of the "object words" are registered in the knowledge base. Based on this, the system associated words and expanded its knowledge within the range of common sense, making possible to handle many expressions. The word association was realized by using the huge Concept Base[6][7] that was automatically built from multiple digital dictionaries. A method to calculate the Degree of Association[8] evaluates the relationship between words. Hereafter, this Concept Base and the calculation method are called the "Association Mechanism".

3 Elemental Technique

3.1 Concept Base

The Concept Base is a large-scale database that is constructed both manually and automatically using words from multiple electronic dictionaries as concepts and independent words in the explanations under the entry words as concept attributes. In the present research, a Concept Base containing approximately 90,000 concepts was used, in which auto-refining processing was carried out after the base had been manually constructed. In this processing, attributes considered inappropriate from the standpoint of human sensibility were deleted and necessary attributes were added.

In the Concept Base, Concept A is expressed by Attributes a_i indicating the features and meaning of the concept in relation to a Weight w_i denoting how important an Attribute a_i is in expressing the meaning of Concept A . Assuming that the number of attributes of Concept A is N , Concept A is expressed as indicated below. Here, the Attributes a_i are called Primary Attributes.

$$A = \{(a_1, w_1), (a_2, w_2), \dots, (a_N, w_N)\}$$

Because Primary Attributes a_i of Concept A are taken as the concepts defined in the Concept Base, attributes can be similarly elucidated from a_i . The Attributes a_{ij} of a_i are called Secondary Attributes of Concept A . Figure 3 shows the elements of the Concept "train" expanded as far as Secondary Attributes.

train	train, 0.36	locomotive, 0.21	railroad, 0.10	...	a_i, w_i	Primary Attributes
	train, 0.36	locomotive, 0.21	railroad, 0.10	...	a_{i1}, w_{i1}	
	locomotive, 0.21	streetcar, 0.23	subway, 0.25	...	a_{i2}, w_{i2}	
	:	:	:	:	:	
	a_{lj}, w_{lj}	a_{2j}, w_{2j}	a_{3j}, w_{3j}	...	a_{ij}, w_{ij}	
						Secondary Attributes

Concept

Fig. 3. Example of the Concept gtrainh expanded as far as Secondary Attributes

3.2 Degree of Association Algorithm

For Concepts A and B with Primary Attributes a_i and b_i and Weights u_i and v_j , if the numbers of attributes are L and M , respectively ($L \leq M$), the concepts can be expressed as follows:

$$\begin{aligned} A &= ((a_1, u_1), (a_2, u_2), \dots, (a_L, u_L)) \\ B &= ((b_1, v_1), (b_2, v_2), \dots, (b_M, v_M)) \end{aligned}$$

The Degree of Identity $I(A, B)$ between Concepts A and B is defined as follows (the sum of the weights of the various concepts is normalized to 1):

$$I(A, B) = \sum_{a_i=b_j} \min(u_i, v_j)$$

The Degree of Association is calculated by calculating the Degree of Identity for all of the targeted Primary Attribute combinations and then determining the correspondence between Primary Attributes. Specifically, priority is given to determining the correspondence between matching Primary Attributes. For Primary Attributes that do not match, the correspondence between Primary Attributes is determined so as to maximize the total degree of matching. Using the degree of matching, it is possible to give consideration to the Degree of Association even for Primary Attributes that do not match perfectly.

When the correspondences are thus determined, the Degree of Association $R(A, B)$ between Concepts A and B is as follows:

$$R(A, B) = \sum_{i=1}^L I(a_i, b_{xi})(u_i + v_{xi}) \times \{\min(u_i, v_{xi}) / \max(u_i, v_{xi})\} / 2$$

In other words, the Degree of Association is proportional to the Degree of Identity of the corresponding Primary Attributes, and the average of the weights of those attributes and the weight ratios.

3.3 Sense Judgment System[4,5]

The knowledge base for the sense and perception judgments has a structure like a thesaurus. It contains sense and perception words that are associated with typical nouns, which have been entered manually. In cases when an unknown word was not registered in the Sense Knowledge Base, when it needs to be processed, the system calculates the Degree of Association with those known words registered in the knowledge base for the sense and perception judgments. Then it chooses the one with the highest Degree of Association for processing. This makes the system obtain the rough corresponding sense and perception. In addition, the system refers to the attributes register in the Concept Base to find the sense and perception particular to that word. Due to its structure, these attributes in the Concept Base contain some inappropriate words as senses and perceptions to be associated. Thus, this system is carefully designed so correct sense and perception is selected by using the Degree of Association.

4 Proposed Emotion Judgment Method

In the existing emotion judgment method, a sentential actor's ("subject") emotion was judged based on the "object words", and "action words". With respect to generated emotions, those associated with a total of 406 pairs of the meaning categories of object words (203 categories). The action words (2 categories of "succession" and "opposite") were manually defined and registered in the knowledge base. However, emotions might not be able to be judged from such processing accurately because polysemy of the word. For instance, sentence "I hit a single" and "I hit a wall" are expressed using same verb. The former is glad, the latter is sadness. Thus, when meaning categories of object words and action words are separately judged, the polysemy cannot be interpreted. In addition, a lot of resources are needed by making exist emotion judgment method and to maintain the knowledge base.

Therefore, a method does not process separately meaning categories of object words and action words, but using both semantic feature of object words and action words is proposed in this paper. Concretely, an utterance sentence translated into a word that expresses semantic feature of object words and action words. Then, the word is classified according to the meaning categories of object words. In addition, emotions are judged using only a meaning category of object words. As a result, both a polysemy problem and the maintenance problem of the knowledge base are solved.

4.1 Proposed Method of Translating an Utterance Sentence into a Word

Proposed method judges a word and it shows a meaning of an utterance sentence consisting by object word and action word. Then, emotions are judged from the category of the word.

Figure 4 shows the outline of the proposed method which translates an utterance sentence into a word. Noun words, which are related to the action word in the

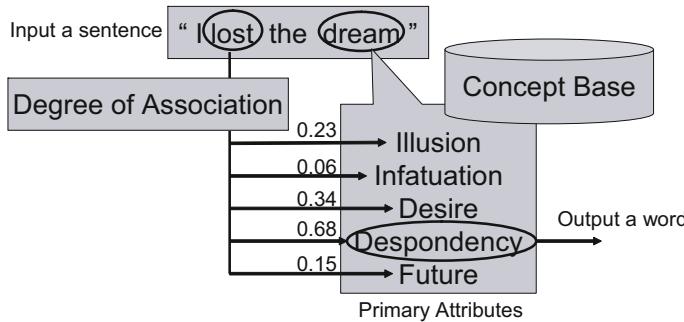


Fig. 4. Outline of conversion from a sentence to a word

sentence are selected from attributes using the Concept Base which mentioned in section 3.1. Then, the Degree of Association which mentioned in section 3.2 is calculated between the noun words and the object word in the sentence. The noun word having the maximal value in the Degree of Association, that shows the meaning of the sentence. Moreover, emotions are judged from the category of the word which judged by same method as the existing method.

4.2 Rebuilding of a Knowledge Base

Emotions are judged from the object words and the action words. With respect to the generated emotions, they associated with a total of 406 pairs of the meaning categories contains object words (203 categories) and action words (2 categories of "succession" and "opposite"), and they were manually defined and registered in the knowledge base. However, a lot of resources are needed for the existing method to make and to maintain the knowledge base.

In the proposed method, emotions are judged from only the meaning categories of object words (203 categories). As a result, a big cost reduction is achieved.

4.3 Selecting of the Meaning Categories Used for Emotions Judgement

Figure 5 shows the outline of the proposed emotion judgement method. In the existing method, object words and action words are categorized to each category. However, in the proposed method, used category is selected by the category of the action words that mentioned in chapter 2. In the case of category "succession[C]" of action words, emotions are judged from the object words that contain in meaning of category. In the other cases of category "positive[A]", "negative[B]" and "opposite[D]", their emotions are judged by the proposed method that mentioned in section 4.1.

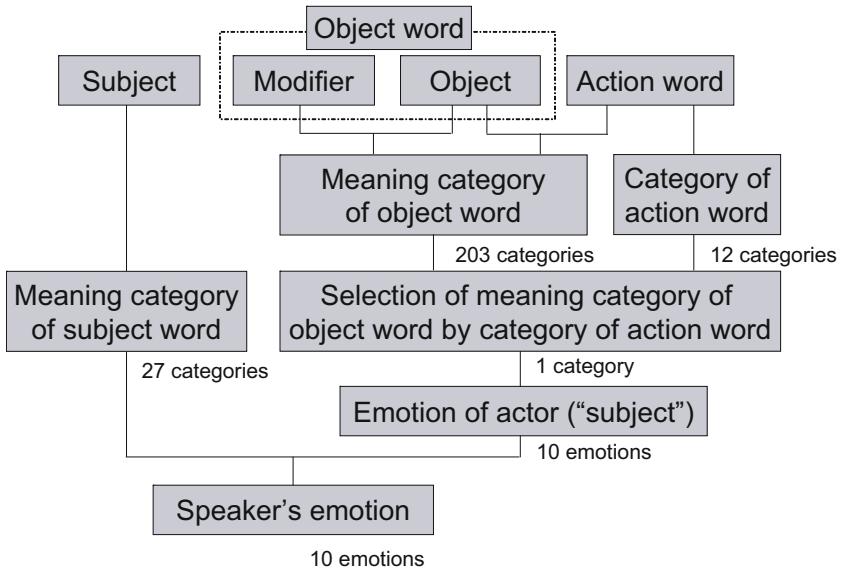


Fig. 5. Outline of proposed Emotion Judgment System

5 Performance Evaluation of the Proposed Emotion Judgment Method

In order to evaluate how valid emotions are generated by proposed emotion judgment method, and 200 sentences were collected from 5 test subjects for using as data for evaluation. Then, five test subjects were then asked to judge whether the emotions generated by the proposed method are common sense. If four or more of judges say generated emotion was a common sense emotion, the emotion is considered as "common sense" (correct answer). In cases when two or three subjects say the emotion was "common sense", the generated emotion is considered for being "not out-of-common-sense". If one subject or no subjects considered the generated emotion as "common sense", the emotion was thought of as "out of common sense" (an error). For cases when multiple emotions were generated, they consider as "common sense" (correct answer) if all of the generated emotions were common sense ones. If any one of the emotions consider being "out of common sense", particular generated emotion consider being "out of common sense" (an error). All others are regarded as "not out-of-common-sense".

Figure 6 shows the results of the proposed emotion judgment method and the existing emotion judgement method. As a result, the proposed method gave the correct answer in 52.9% of cases. If the "not out-of-common-sense" answers were counted as part of the "correct answers", the correct-answer ratio rose up to 78.9%. The accuracy of the proposed emotion judgment method was improved 18.4% by comparing with the existing method. By the results of emotion judgement were improved, we believe that the proposed method is effective. However,

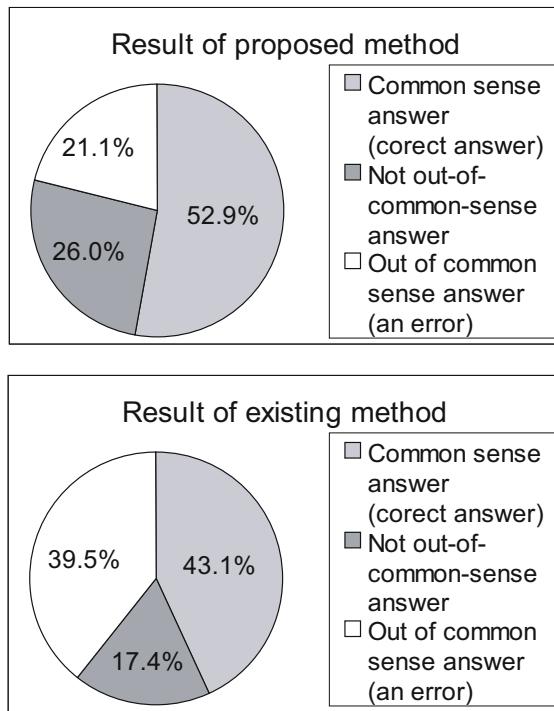


Fig. 6. Result of proposed Emotion Judgment method

metaphor expressions cannot be correctly interpreted by the proposed method. It will be necessary to interpret such expressions in the future.

6 Conclusions

Authors focus on the emotion of such common sense and attempt to compose a method to judge the user's emotions based on utterances. Such system and method have already been developed. However, emotions might not be able to judge from the existing method accurately because of the polysemy of the word. In addition, a lot of resources are needed for the existing emotion judgment method to make and maintain a knowledge base. Therefore, a method was not processing meaning categories of words separately, but translating an utterance sentence into a word was proposed in this paper. Herewith, both a polysemy problem and the maintenance problem of the knowledge base could be solved. The proposed method used knowledge base and an Association Mechanism.

As a result, the proposed method gave the correct answer in 52.9% of cases. The accuracy of the proposed emotion judgment method was improved approximately 18.4% compared with an existing method. By the results of emotion judgement were improved, we believe that the proposed method is effective.

Acknowledgment. This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 21700241).

References

1. Tsuchiya, S., Yoshimura, E., Watabe, H., Kawaoka, T.: The Method of the Emotion Judgment Based on an Association Mechanism. *Journal of Natural Language Processing* 14(3), 119–238 (2007)
2. Tsuchiya, S., Yoshimura, E., Ren, F., Watabe, H.: Emotion Judgment based on Relationship between Speaker and Sentential Actor. In: Velásquez, J.D., Ríos, S.A., Howlett, R.J., Jain, L.C. (eds.) *KES 2009. LNCS (LNAI)*, vol. 5711, pp. 62–69. Springer, Heidelberg (2009)
3. NTT Communication Science Laboratory: *NIHONGOGOITAIKEI*. Iwanami Shoten (1997) (Japanese)
4. Horiguchi, A., Tsuchiya, S., Kojima, K., Watabe, H., Kawaoka, T.: Constructing a Sensuous Judgement System Based on Conceptual Processing. In: Gelbukh, A. (ed.) *CICLing 2002. LNCS*, vol. 2276, pp. 86–95. Springer, Heidelberg (2002)
5. Watabe, H., Horiguchi, A., Kawaoka, T.: A Sense Retrieving Method from a Noun for the Commonsense Feeling Judgement System. *Journal of Artificial Intelligence* 19(2), 73–82 (2004)
6. Hirose, T., Watabe, H., Kawaoka, T.: Automatic Refinement Method of Concept-base Considering the Rule between Concepts and Frequency of Appearance as an Attribute. Technical Report of the Institute of Electronics, Information and Communication Engineers. *NLC 2001-93*, pp. 109–116 (2002)
7. Kojima, K., Watabe, H., Kawaoka, T.: A Method of a Concept-base Construction for an Association System: Deciding Attribute Weights Based on the Degree of Attribute Reliability. *Journal of Natural Language Processing* 9(5), 93–110 (2002)
8. Watabe, H., Kawaoka, T.: Measuring Degree of Association between Concepts for Commonsense Judgements. *Journal of Natural Language Processing* 8(2), 39–54 (2001)

A Computationally Efficient Fuzzy Logic Parameterisation System for Computer Games

Leslie Jones, Robert Cox, and Sharifa Alghowinem

Faculty of Information Systems and Engineering,
University of Canberra,
ACT 2610, Australia

{Leslie.Jones,Robert.Cox}@canberra.edu.au,
sharifa.m.f@gmail.com

Abstract. Linguistic fuzzy expert systems provide useful tools for the implementation of Artificial Intelligence (AI) systems for computer games. However, in games where a large number of fuzzy agents are needed, the computational needs of the fuzzy expert system inclines designers to abandon this promising technique in favour of non-fuzzy AI techniques with a lower computational overhead. In this paper we investigated a parameterisation of fuzzy sets with the goal of finding fuzzy systems that have lower computational needs but still have sufficient accuracy for use in the domain of computer games. We developed a system we call short-cut fuzzy logic that has low computational needs and seems to have adequate accuracy for the games domain.

Keywords: Fuzzy Logic, Fuzzy Control, Computer Games, Computational Efficiency.

1 Introduction

The use of linguistic fuzzy logic in computer games can help increase the quality of interaction in games [1]. Linguistic fuzzy logic rules can be quite human-readable. In fact if done with a view to human readability, such systems can be constructed so that the logic is quite human like and can provide a useful tool for the creation of a wide range of artificial intelligence systems suitable for computer games. The use of fuzzy logic rules of this nature helps to overcome the shortcomings of classical logic when dealing with imprecise linguistic semantics [2].

Although fuzzy logic overcomes linguistic issues, computational times become a critical issue [3] which has an impact on games attempting to compute AI responses for multiple agents within a short time frame. Chen [3] reports two types of techniques to reduce the computational time, using singletons and table look-up in the if-part and using mathematical equations in the then-part. In this paper we present a parameterised fuzzy set representation as our solution to reducing computational times.

There are many applications of fuzzy logic in games. Bourg and Seemann [4] give three examples of application; controlling bots, assessing threats to bots and classifying bots. For our paper we consider controlling the movement of large numbers of bots and the avoidance of obstacles, both moving and static.

Linguistic fuzzy rules take the form of *IF antecedent THEN consequent*. The simple example rule below is from a wall-avoidance system in our test system, and shows how this syntax provides a useful way of converting expert human knowledge into an AI system.

```
IF right-distance IS close AND top-distance IS closish
    THEN speed IS slow AND turn IS sharp-right
```

2 Hypothesis

Given that there are many possible parameterisations of fuzzy logic systems our hypothesis is that the parameterisation proposed here has enough accuracy and performance for computer games involving large numbers of agents. Our goal is to test that is both computationally efficient and also preserves the human-readable form of linguistic fuzzy logic. We take 'computationally efficient' to mean a parameterisation that minimises the use of real numbers and division operators, and also avoids computations using arrays. By selecting a parameterisation that has high efficiency designers of computer games will be able to increase the sophistication or number of PVE (player versus environment) bots without requiring vast amounts of processing power.

3 Design of Our Parameterisation

Fuzzy systems are built around evaluating rule sets having the general form 'IF antecedent THEN consequent'. To evaluate these rule sets efficiently means that we need a computationally light antecedent and a computationally light consequent. In terms of consequent, Sugeno [5] proposed and demonstrated a low computational cost consequent system using fuzzy singletons. Zarozinski [6] used lookup tables to reduce consequent computational cost. In this paper we chose fuzzy singletons for the consequent. Only the antecedent part of the equation is considered for simplification in this paper.

We note that others [7] have produced fast fuzzy controllers which depend on using a Field Programmable Gate Array (FPGA). However our target area is PC or Console based games, enabling the system proposed here to work on generic varieties of equipment available to game players.

Our final design is a simplified fuzzy set which we call a Short-cut fuzzy (SCF) set with just seven parameters: Width at Height (WH), Height (H), Width at Base (WB), Centre (C), Base (B), World Constant (minimum world value) (WC), World Scale (maximum world value) (WS) (see Fig. 1). As a reference set we use a symmetrical isosceles trapezoid. By adjusting the parameters we can create a range of set shapes such as rectangular, triangular, vertical and horizontal lines. We can also create truncated versions of these set shapes.

With our SCF parameterisation it is not possible to create the traditional operators AND and OR [8] that function on our SCF sets. However, by using Sugeno's method for the consequent there is no requirement to perform AND and OR on SCF sets only on singletons (normal Sugeno accumulator). Since in the antecedent, AND and OR can be done after fuzzification on the fuzzified values, then there is no need to AND

and OR the fuzzy sets. In most fuzzy systems, as described in Negnevitsky [9], the antecedent sets tend to be relatively straight forward meaning that in many cases there is no computational consequence in using our short-cut sets. Therefore it is possible to create intricate fuzzy systems which do not use AND and OR on the fuzzy sets.

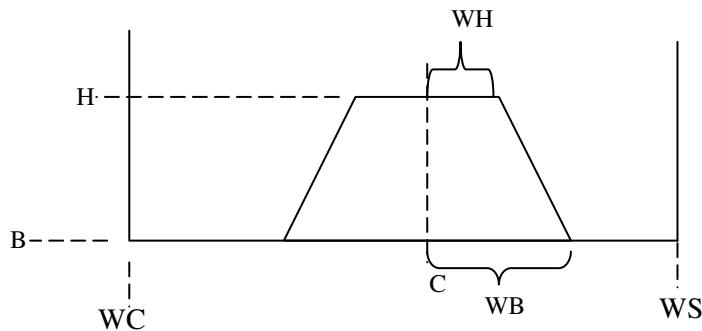


Fig. 1. Visual Representation of Our Parameterisation

Our short-cut system needs to be compared against a more traditional system. We use Mamdani-style inference [9] (using piecewise linear functions) where an equivalent fuzzy set would have a 12 parameter representation. The 12 parameters would be WC, WS, the y coordinates for WC and WS, and the (x, y) coordinates for points 1, 2, 3 and 4 (see Fig. 2). We are aware that a very small set such as a single line would have less parameters than the system we are proposing. However in general, piecewise linear sets would use more parameters than the short-cut fuzzy sets we are proposing.

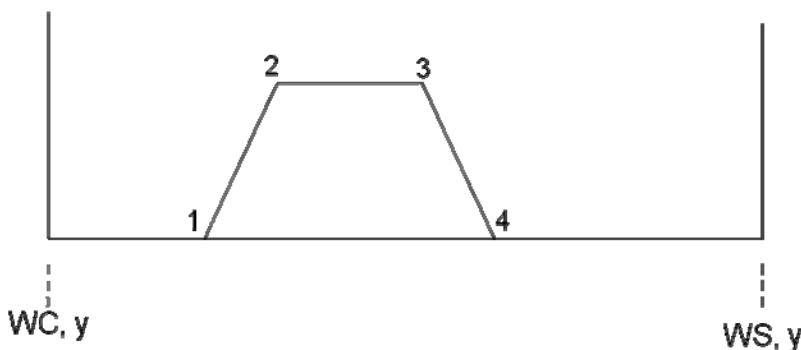


Fig. 2. Visual Representation of a Piecewise Linear Set

4 Method

The broad methodology was to construct a fairly traditional collision avoidance system using piecewise linear fuzzy sets and hand-optimise it without regard to set shape (i.e. to build the best system we could). The collision avoidance system we implemented is similar to that used by Gerdelen & Reyes [10] in that we used three ranges when determining turning behaviour. We then converted the system to use the lower computational cost short-cut system. In this conversion all fuzzy rules were retained and only the fuzzy sets changed.

4.1 The Experiments

We ran two sets of experiments. For each set we increased the number of moving objects from 25 to 150 in steps of 25. We performed 40 runs each of 500 turns creating a new set of random obstacles for each run. For each set we kept the number of static object constant - 10 for the first set and 40 for the second.

We used three measures to compare the accuracy of each system:

- Collision with static objects,
- Collision with moving objects, and
- Collisions with game-edge boundaries (referred to hereafter as walls).

We used two measures to compare the efficiency of each system:

- Time in seconds to complete the 500 turns,
- Average frames per second (FPS) for the runs

We wanted to simulate the ‘hustle and bustle’ of a typical RTS (Real Time Strategy) game hence the use of a multi agent system (see Fig. 3).

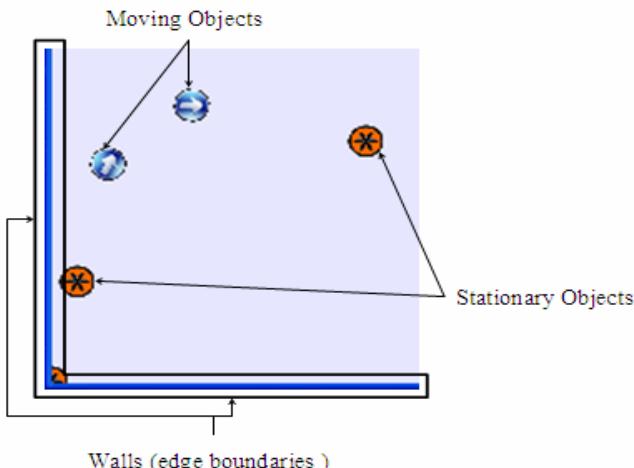


Fig. 3. Screen Shot Showing the Agents in the System

Both our baseline model implementation and the SCF implementation used a Sugeno accumulator consequent with defuzzification also using the Sugeno method.

We aimed to show that the short-cut fuzzy implementation would use less computational time such that when used in mass agent control systems performance (i.e. frame rate) would be enhanced without creating a significant loss in accuracy, which is a desired outcome for fuzzy control of agents in computer games.

4.2 Construction of the System

To test our hypothesis, that the short-cut fuzzy parameter sets allow agents to be constructed that are more computationally efficient in large numbers, we built a complex system simulating the movement of multiple sprites that needed to avoid collision with each other. The baseline model was constructed using traditional fuzzy logic with piecewise linear parameterisation requiring (typically) 12 parameters. We then optimised the system without regard to the shape of the fuzzy sets as if it was a traditional fuzzy controller. The system and its sets were created manually using trial and error until an acceptable result was achieved. The test program counted the number of collisions for a given number of movement events (in our case 500 turns), each turn of which involved a movement event for each agent in the simulation.

Set Replacement Process. To implement the SCF model we replaced the final piecewise linear sets with similar short-cut fuzzy sets. Since short-cut fuzzy sets do not cover the same range of possibilities as normal fuzzy sets, the resultant sets were not always identical. For example, a normal asymmetrical triangular set was replaced by an isosceles triangle set in the short-cut implementation (see Fig. 4). Note that in normal circumstances you would use SCF sets to create your system from the start – no replacement process would be necessary.

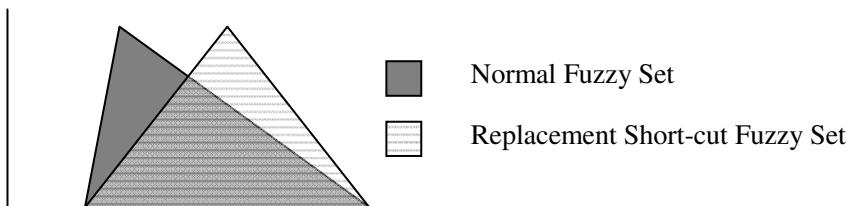


Fig. 4. Single Replacement from Normal to Short-cut Fuzzy Set

However, several methods could be used to overcome this kind of conversion, such as creating two equilateral triangles to reduce the error rate (see Fig. 5).

Rules. Our fuzzy rule sets were grouped into two categories. Firstly, the object-avoidance rules controlled how moving agents reacted to other moving and stationary agents at three different ranges (very close, close and far) to avoid such objects. Secondly, the wall-avoidance rules controlled how moving agents reacted to the walls and corners of the environment by moving the agent towards the centre of the nearest wall and away from the corner. Both sets of rules were used for the traditional and the short-cut implementations without modification.

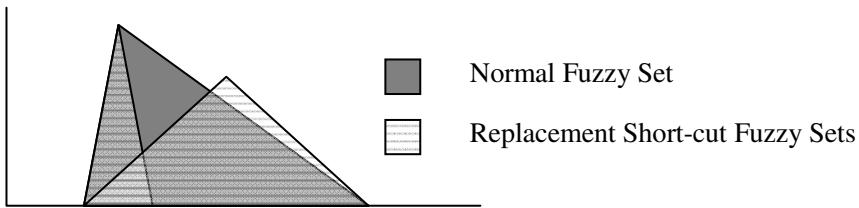


Fig. 5. Multiple Replacement from Normal to Short-cut Fuzzy Sets

Object-avoidance Rules.

```

IF distance IS very close AND angle IS far Left
  THEN speed IS slow AND direction IS sharp right
IF distance IS very close AND angle IS far Right
  THEN speed IS slow AND direction IS sharp left

IF distance IS close AND angle IS close Left
  THEN speed IS mod AND direction IS right
IF distance IS close AND angle IS close Right
  THEN speed IS mod AND direction IS left

IF distance IS far AND angle IS dangerous Left
  THEN speed IS fast AND direction IS slow right
IF distance IS far AND angle IS dangerous Right
  THEN speed IS fast AND direction IS slow left

```

Wall-avoidance Rules.

```

IF right distance IS close AND top distance IS closish
  THEN speed IS slow turn sharp right
IF right distance IS close AND bottom distance IS
closish
  THEN speed IS slow turn sharp left

IF left distance IS close AND top distance IS closish
  THEN speed IS slow turn sharp left
IF left distance IS close AND bottom distance IS
closish
  THEN speed IS slow turn sharp right

```

```

IF top distance IS close AND right distance IS closish
THEN speed IS slow turn sharp left
IF top distance IS close AND left distance IS closish
THEN speed IS slow turn sharp right

IF bottom distance IS close right distance IS closish
THEN speed IS slow turn sharp left
IF bottom distance IS close left distance IS closish
THEN speed IS slow turn sharp right

```

Note: after processing all rules, the possibility of a collision still exists, since rule interactions of moving objects with stationary objects and walls can still result in a movement vector that results in a collision. It is these collisions that we count. When such a collision occurs, any moving agents involved are stopped and rotated until a "safe" direction is found then they continue moving.

5 Results

All statistical tests were two tailed t-tests for two samples assuming equal variances and $p=0.05$.

For the 10 static objects test set there were no significant differences in the number of static and moving collisions for the full range of moving objects from 25 to 150 objects. For run time and average frame rate there were no significant differences for 25 to 75 moving objects. There were significantly better results for the SCF sets in run time and average frame rate for 100, 125 and 150 moving objects.

Likewise for the 40 static objects test set there were no significant differences in the number of static and moving collisions for the full range of moving objects from 25 to 150 objects. For run time and average frame rate there were no significant differences for 25 to 75 moving objects. There were significantly better results for the SCF sets in run time and average frame rate for 100, 125 and 150 moving objects.

There was a trend for the number of wall collisions to be significantly higher for the normal implementation declining in significance up to 100 moving objects. While the chart 'Wall Collisions' in Figure 6 does not make this obvious the t-test does show the significant differences.

Interestingly there were no significant differences in the number of moving and wall collisions between any combination of normal or short-cut technique and 10 or 40 static objects. There was a significant difference in the number of static collisions between the 10 and 40 static object runs.

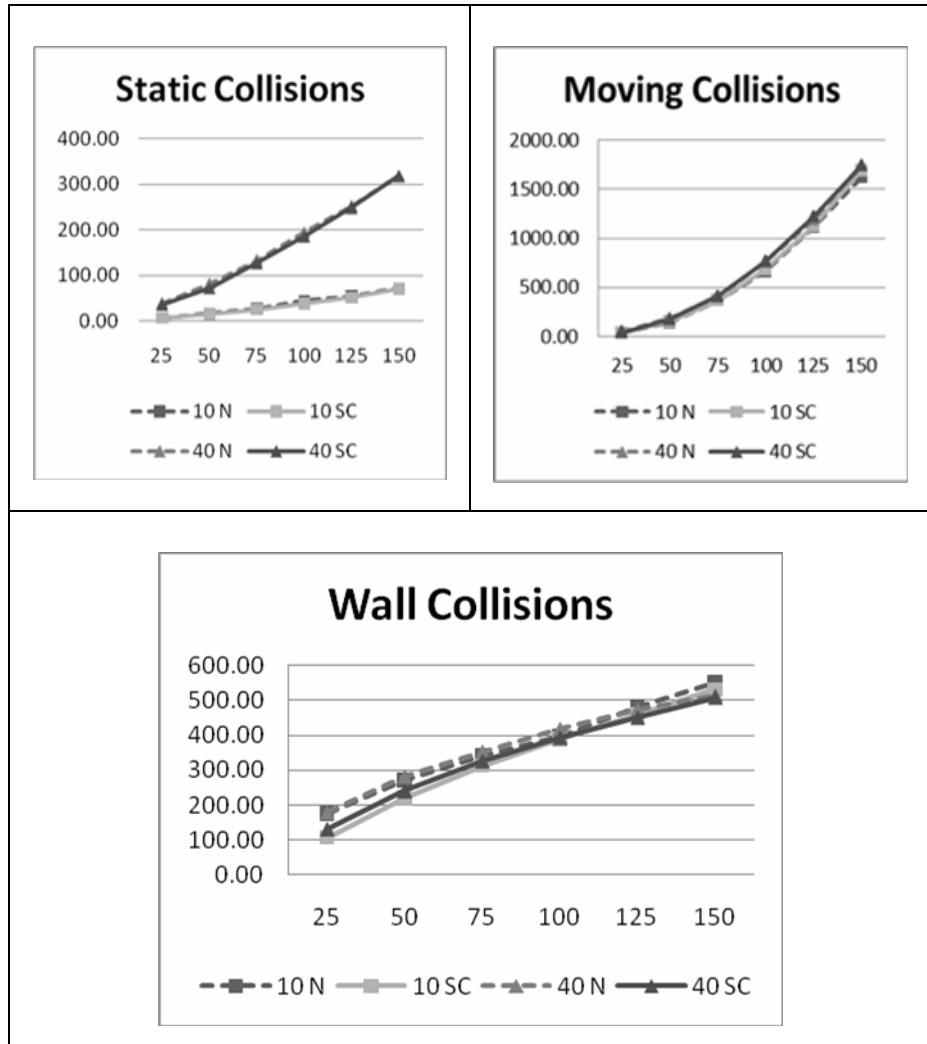


Fig. 6. Charts of Collision Results

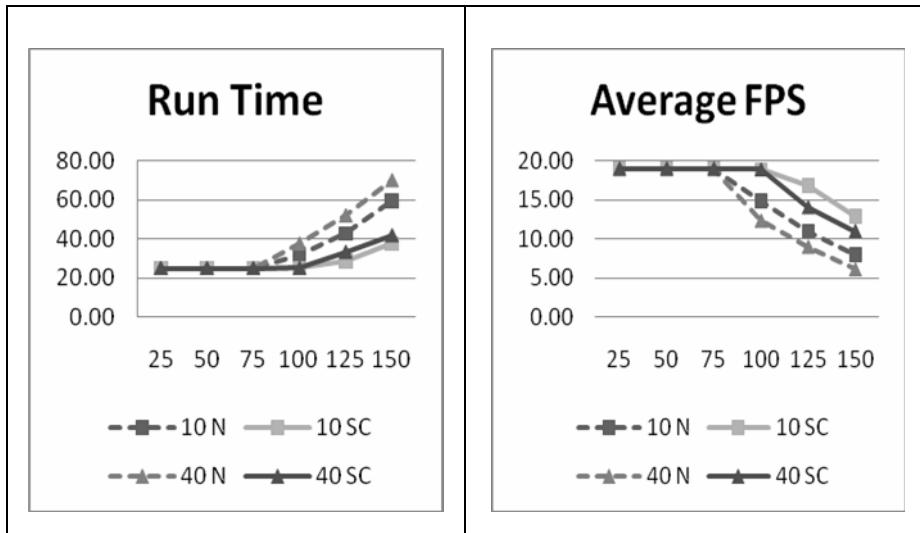


Fig. 7. Charts of run time and Average FPS results

6 Analysis of Results

The results support two main points. Firstly, there was no significant loss of accuracy in using short-cut fuzzy sets as opposed to normal sets in the design of the testing system no matter how many moving objects we used. Secondly, there was a significant loss of performance for normal fuzzy sets when used with large numbers of moving objects (from 100 onward).

These two points support our hypothesis that a parameterisation of fuzzy sets exists that maintains performance without major loss of accuracy when used with large numbers of agents. In fact the SCF set implementation using both 10 and 40 static objects performed better than both normal implementations.

The significant difference in wall collisions was only apparent for low numbers of agents and does not refute our hypothesis. It is entirely consistent with the set replacement process as described in section 4.2. The SCF sets in this case seem to be slightly better than the ones in the normal implementation.

In the results we noted that with 40 static objects the number of moving and wall collisions was similar but the static collisions changed significantly. Two explanations come to mind. Firstly, the number of static objects has increased and the increase is enough to create more static collisions but not enough to cause a significant decrease in moving and wall collisions. Secondly, the rules for avoiding static and moving objects are evaluated at the same time, however, with two moving objects both are trying to avoid a collision whereas with a static and moving object only the moving object is trying to avoid a collision. So with more static objects, it becomes harder to avoid a static collision. If we increase the number of static objects greatly the screen will become so congested that even moving and wall collisions will be significantly less avoidable.

7 Conclusion

The results indicate that fuzzy systems can be built with fuzzy sets that are limited to the parameterisation we propose. Clearly, such systems do not embody the same range of functionality that a fully parameterised system would permit. However, as discussed in the analysis, the short-cut system we propose is computationally faster, as shown by the better run times and average frame rates when operating with large numbers of agents. We suspect that for RTS games with large numbers of agents, these savings would be significant.

8 Further Research

It became apparent that with only five parameters that require optimisation, such sets offer the prospect for machine optimisation and are better suited for machine optimisation than the piecewise linear counterpart.

References

1. Li, Y., Musilek, P., Wyard-Scott, L.: Fuzzy Logic in Agent-Based Game Design. Annual Meeting of the North American Fuzzy Information Processing Society 2, 734–739 (2004)
2. Zadeh, L.A.: Fuzzy logic. Computer 21(4), 83–93 (1988)
3. Chen, H.P., Shyur, J.C., Parng, T.M.: Toward Fast Reasoning for Fuzzy Logic Inference. In: Proceedings of 1993 International Joint Conference on Neural Networks, IJCNN 1993, vol. 1, pp. 697–700 (1993)
4. Bourg, D.M., Seemann, G.: Fuzzy Logic. In: AI for Game Developers, pp. 188–211. O'Reilly, Sebastopol (2004)
5. Takagi, T., Sugeno, M.: Fuzzy Identification of Systems and Its Applications to Modelling and Control. IEEE Transactions on Systems, Man and Cybernetics 15, 116–132 (1985)
6. Zarozinski, M.: An Open-Source Fuzzy Logic Library. In: Rabin, S. (ed.) AI Game Programming Wisdom, pp. 90–101. Charles River Media, Hingham (2002)
7. Deliparaschos, K.M., Nenedakis, F.I., Tzafestas, S.G.: A Fast Digital Fuzzy Logic Controller: FPGA Design and Implementation. In: 10th IEEE Conference on Emerging Technologies and Factory Automation, ETFA 2005, vol. 1, pp. 259–262 (2005)
8. Zadeh, L.A.: Fuzzy Sets. Information and Control 8, 338–353 (1965)
9. Negnevitsky, M.: Fuzzy Expert Systems. In: Artificial Intelligence: A Guide to Intelligent Systems, 2nd edn., Addison Wesley, Harlow (2005)
10. Gerdelen, A.P., Reyes, N.H.: A Novel Hybrid Fuzzy A* Robot Navigation System for Target Pursuit and Obstacle Avoidance. In: Proceedings of the First Korean-New Zealand Joint Workshop on Advance of Computational Intelligence Methods and Applications, vol. 1, pp. 75–79 (2006)

Representation and Reuse of Design Knowledge: An Application for Sales Call Support

Julian R. Eichhoff¹ and Wolfgang Maass^{1,2}

¹ Furtwangen University, Robert-Gerwig-Platz 1, 78120 Furtwangen, Germany
`{julian.eichhoff,wolfgang.maass}@hs-furtwangen.de`

² University of St. Gallen, Dufourstrasse 40a, 9000 St. Gallen, Switzerland
`wolfgang.maass@unisg.ch`

Abstract. This paper presents a Bayesian network representation of the function-behavior-structure (FBS) framework [10], which is used to guide salespersons through conceptual design tasks in lead qualification situations. After we outline the lead qualification situation and state the need of design support for salespersons, a review of the related works shows the necessity for a knowledge representation, which explicitly addresses the uncertainty of design decisions. In the remainder we propose a representation, which is capable of this, and close with an application example for sales call support.

Keywords: Knowledge modelling, Bayesian networks, design computing, conceptual design, lead qualification, function-behavior-structure framework.

1 Introduction

Increasing customer-oriented project delivery and implementation has been recognized as an important change in the context of organizational buying (B-to-B) [13]. For industries that offer customized solutions, fluid less structured knowledge is important for getting a shared understanding between customers and vendors. Typical examples of highly customer-oriented projects are office fit-out projects, as they may have an substantial impact on the customer's business operations [20]. They deal with the design and construction of the scenery and settings of office accommodation, aligned with the customer's very own aims, needs, structure and identity [2]. In the very first project phase, called lead qualification, the sales force is required to evaluate the readiness, willingness, and ability of a customer to buy an offer. With increasing customer-orientation this changes to a consultative "solution selling" task [18], demanding creative problem-solving skills [26], or in other terms conceptual designing experience. In this paper we propose a computational representation of office design knowledge and a method for knowledge reuse, to efficiently guide salespersons through the conceptual design processes in lead qualification situations for office fit-out projects.

2 Problem Statement and Approach

Lead qualification can be conceptualized as a principal-agent situation, where the vendor delegates the tasks carried out in lead qualification to a salesperson (cf. [4]). The beginning of lead qualification is characterized by an information asymmetry that reflects a situation in which the company has insufficient information about what the lead (potential customer) desires [25]. The salesperson acts as an mediator establishing a form of corporate communication between lead and vendor to reduce information asymmetry. This is mostly done iteratively in several sales calls with representatives of the lead. The salesperson uses the meetings to identify problems of the lead that can be solved by the vendor's goods and services. These perceptions are then reported back to the vendor where the information is used to evaluate the chances for business, and, if necessary, prepare adequate offerings. Since the accuracy of these perceptions depends only on the salesperson himself, his individual performance critically determines the outcome of the vendor's consultancy efforts [12][22].

A crucial issue in this setting is given by salespersons asking the wrong questions, i.e. they fail to gather right information in sales calls and thus miss important business opportunities. Following the notion that a solution is not only a mere consequence of a stated problem but also helps to (re-)structure a problem [6], salespersons should be aware of possible solutions to efficiently interview leads about their business needs. This in turn requires sophisticated conceptual design knowledge considering the characteristics of a solution (cf. [24]).

Our approach is to provide mechanisms to formalize the required design knowledge and reuse it by means of a dynamic questionnaire that will adapt to the current lead qualification situation when provided with answers. Considering the information that has already been gathered on the lead, the design knowledge is used to estimate a current state of the problem/solution space as seen from the vendor's point of view. Given this estimation we highlight those questions that are most insightful in the current situation. The answers provided by the salesperson are then used to restructure the problem/solution space and in consequence highlight consecutive questions. Embedded in an information system (IS) the questionnaire will guide the salesperson's preparations for sales calls.

3 Related Work

To the best of our knowledge, there exist no systems that explicitly support the lead qualification for office fit-out projects. But viewing the problem from a lead's perspective there have been several approaches that deal with the problem of contractor pre-qualification, i.e. the screening for capable vendors (cf. [8]). Furthermore, looking at the domain of project tendering, which can be seen as a downstream process to lead qualification, different decision support systems have been proposed to assist the vendor in estimating whether it is feasible or not to tender, the so called bid/no-bid problem (cf. [17]). Both are formulated as classification problems trying to measure the fitting of the lead's problems

with the vendor's problem solving capabilities. But reconsidering the solution selling aspect, an adequate model for lead-qualification should not only allow for classifying different states of the problem/solution space but also be capable of generating this space.

A promising modeling approach comes from the field of design research: Functional concept ontologies provide a holistic view on the problem/solution space and may be used to assist people to deal with the complexity of conceptual design [3]. For a comprehensive review see [9]. The (situated) function-behavior-structure (FBS) framework [10] seems especially appropriate, as it focusses on the design object generation processes and intrinsically supports the notion of demands and offers. Even though the authors do not propose a formalized methodology to decompose the functions or to associate the functions with behaviors and structure [9], other works implemented computational representations of the FBS framework. In [15] a UML class diagram scheme of the FBS model is used to represent the interrelations of processes, products, resources and external effects in product life-cycles. [5] took a similar entity-relationship approach and provided an ontological FBS representation for conceptual design. Other approaches of functional concept ontologies have been implemented by means of some notion of a state transition system (e.g. [11]), where the state space provides a configurational description of the design object. Further, models for designing are defined to put constraints on the operations carried out in the state space, i.e. state transitions and production/association rules for design object entities (e.g. [27][28]).

However, none of these models provide a formal mechanism to specifically address the uncertainty of a design decision. Common theories of the management of uncertainty in design decisions are reviewed in [19]. The authors suggest probability theory as an appropriate approach, given the premise that probabilities for the outcomes of different design decisions can be defined from data (objective probability) or judgement (subjective probability). I.e. relevant design concepts are conceived as random variables, which define a state space whose realizations are more or less probable with respect to the objective frequency of past observations or to the subjective beliefs of an individual. Bayesian networks (BN) provide a well known conceptual framework to integrate multiple random variables to form a dependency network of conditional probabilities. These conditional probabilities are often used to express causal relations between concepts [21], likewise the associations inherent in a FBS model. Beside more general applications for information retrieval [7] BNs have been applied for design reasoning [16][23].

The idea of reasoning from a functional concept ontology defined in form of a BN shouldn't be seen as counterintuitive to the idea of case-based reasoning (CBR) [1], but as "soft computing" component in the technology stack for hybrid intelligent (design) systems (cf. [29]). In fact it can be used to implement the retrieve, reuse, revise and retain steps, as shown in [1]. Their BN-based CBR implementation not only considers experience from previous cases by using data mining (objective probabilities). It also integrates human generated design beliefs defined in domain ontologies (subjective probabilities).

In the remainder of the paper we present a novel BN-approach to operationalize the FBS framework, specially designed to cope with the uncertainty inherent in the vendor's view on the problem/solution space.

4 Representation of Office Design Knowledge

In the FBS framework a design object is described by (ranges of) values for three sets of variables, which define the problem/solution space (cf. [10]): Function (F) variables "describe the teleology of the object, i.e. what it is for" [10]. To cast this notion of Function in to the domain of office fit-out projects one should consider the project's value proposition. From the lead's perspective two factors contribute to this value proposition, the generation of benefits (e.g. flexibility to deal with changes in staff personnel, or represent corporate image) and the avoidance of costs (e.g. reduce vacancy rates, or lower operating costs) (cf. [2]). Structure (S) variables describe the components used for implementation. Regarding an office fit-out project this includes all goods, such as furniture and other interior elements, as well as services, like design, construction and project management, provided to the lead. Behavior (B) variables have a special role as they provide links between Function and Structure variables. Behavior variables are conceptualized as observable attributes that are exhibited by a solution (e.g. storage capability, adjustability of workplaces, degree of privacy). These variables hold two values (or ranges of values). Beside the value that is derived from a given Structure (Bs) representing the vendor's offer, they may also have an expected value (Be) representing the Behaviors demanded by the lead. The latter is derived from the defined Function variables. In this sense the FBS framework provides an integrated view on design objects, combining the problem and solution domain to form a combined space. The act of designing can be represented as a set of operations modifying this space by adding or removing variables and assigning (ranges of) values to the variables.

As mentioned the FBS framework is represented as BN to facilitate its computational use. BNs are instantiated to define the problem/solution space of a specific lead qualification situation. Instantiations are generated upon a predefined template called the FBS Network Template (FBS-NT), which encodes the vendor's design knowledge by means of conditional probabilities. Since BNs are generative probability models, we can compute estimates for all variables in the problem/solution space via Bayesian inference. These probability estimates are conceptualized as a vendor's guess of the problem/solution space given his design knowledge.

In a FBS-NT (cf. Def. 1) Functions, Behaviors and Structures are represented as random variables, which define the nodes of a directed acyclic graph (DAG). All random variables are discrete to simplify the computation of the Bayesian inference later on. Their states define the possible configurations of the problem/solution space. Connections between these variables are defined as conditional probability distributions represented by the graph's edges. Possible relations are $F \rightarrow B$ (Function expects Behavior) and $S \rightarrow B$ (Structure exhibits

Behavior) as well as relations denoting implications within a variable group, i.e. $F \rightarrow F$, $B \rightarrow B$ and $S \rightarrow S$. Further all variables are assigned to a distinct aspect of the problem/solution space (e.g. Project, Business, Office, User), termed Perspective. A Perspective may be related to other Perspectives to express dependencies like “Users can be related to Offices”.

Definition 1 (FBS Network Template). Let $G = (U, E)$ be a directed acyclic graph (DAG), and let $\mathbf{X} = (X_u)_{u \in U}$ be a set of random variables indexed by nodes U , and let $P(\mathbf{X})$ be the joint probability over all variables with edges E representing the conditional dependencies, and let $G_{Pers} = (V_{Pers}, E_{Pers})$ be an undirected graph with nodes V_{Pers} representing Perspectives and edges E_{Pers} expressing *canBeRelatedTo* relations among the Perspectives, then (\mathbf{X}, G_{Pers}) is a FBS Network Template (FBS-NT), given the properties:

Partitioning. Every variable $X \in \mathbf{X}$ is assigned to a Perspective $v \in V_{Pers}$ with $inPerspective : \mathbf{X} \rightarrow V_{Pers}$.

Variables. Let Ω_X be a set of possible states for a discrete random variable X , then:

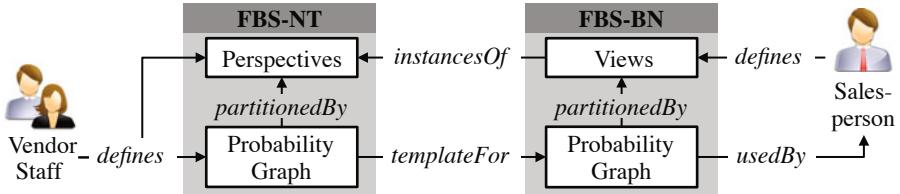
- Every Function is defined as $F : \Omega_F \rightarrow [0, 1] \in \mathbf{F}, \mathbf{X}$.
- Every Behavior is defined as $B : \Omega_B \rightarrow [0, 1] \in \mathbf{B}, \mathbf{X}$.
- Every Structure is defined as $S : \Omega_S \rightarrow [0, 1] \in \mathbf{S}, \mathbf{X}$.

Factorization. Preserving the DAG property of G the joint probability $P(\mathbf{X})$ may be arbitrarily factorized with conditional probabilities $P(X | Pa(X))$ of the following types, where $Pa(X)$ is the set of parents of X :

- $X \in \mathbf{F}$ and $Pa(X) \subseteq \mathbf{F} \setminus X$ (Function implicates Function)
- $X \in \mathbf{B}$ and $Pa(X) \subseteq \mathbf{F}$ (Function expects Behavior)
- $X \in \mathbf{B}$ and $Pa(X) \subseteq \mathbf{B} \setminus X$ (Behavior implicates Behavior)
- $X \in \mathbf{B}$ and $Pa(X) \subseteq \mathbf{S}$ (Structure exhibits Behavior)
- $X \in \mathbf{S}$ and $Pa(X) \subseteq \mathbf{S} \setminus X$ (Structure implicates Structure)

The formalized design knowledge provided by a FBS-NT is used to instantiate a FBS Bayesian Network (FBS-BN, cf. Def. 2) for a specific lead qualification situation (cf. Fig. 1). The Perspectives of the FBS-NT frame the possibilities for instantiation. In a FBS-BN there may be multiple instances of these Perspectives, called Views. Every View stands for a complete duplicate of a Perspective’s variables and their assigned relations, given a slight difference: All variables in B are represented twice in an FBS-BN, i.e. Be variables stand for the expected value of a Behavior, and Bs variables represent the value derived from Structure. Further additional Bc nodes are used to compare the Be and Bs values to measure their match. By defining $P(Bc = true | Be = Bs) \stackrel{\text{def}}{=} 1$ and 0 for all other cases we rigidly couple the problem and the solution space. By means of Bayesian inference, this property allows us to select a Structure that fits a defined Function or vice versa discover the Functions that are provided by a given Structure.

Definition 2 (FBS Bayesian Network). Let $G' = (U', E')$ be a directed acyclic graph (DAG), and let $\mathbf{X}' = (X'_{u'})_{u' \in U'}$ be a set of random variables

**Fig. 1.** Instantiation Concept

indexed by nodes U' , and let $P(\mathbf{X}')$ be the joint probability over all variables with edges E' representing the conditional dependencies, and let $G_{View} = (V_{View}, E_{View})$ be an undirected graph with nodes V_{View} representing Views and edges E_{View} expressing *isRelatedTo* relations among the Views, then (\mathbf{X}', G_{View}) is a FBS Bayesian Network (FBS-BN) with respect to a FBS-NT (\mathbf{X}, G_{Pers}) , given the properties:

Partitioning. Every variable $X' \in \mathbf{X}'$ is assigned to a View $v' \in V_{View}$ with $\text{inView} : \mathbf{X}' \rightarrow V_{View}$.

Instantiation. Every View $v' \in V_{View}$ is assigned to a Perspective $v \in V_{Pers}$ with $\text{instanceOf} : V_{View} \rightarrow V_{Pers}$.

Variables. For every View $v' \in V_{View}$ with $\text{instanceOf}(v') = v$:

- There is a random variable $F' \in \mathbf{F}'$, \mathbf{X}' with $\text{inView}(F') = v'$, where F' is a duplicate of the FBS-NT's Function F with $\text{inPerspective}(F) = v$.
- There are two random variables $Be \in \mathbf{Be}$, \mathbf{X}' and $Bs \in \mathbf{Bs}$, \mathbf{X}' with $\text{inView}(Be) = v'$, $\text{inView}(Bs) = v'$, where Be and Bs are duplicates of the FBS-NT's Behavior B with $\text{inPerspective}(B) = v$. Further there is an extra random variable $Bc : \{\text{true}, \text{false}\} \rightarrow [0, 1] \in \mathbf{X}'$ with $\text{inView}(Bc) = v'$ and $P(Bc | Be, Bs)$, where $Bc = \text{true}$ denotes a match and $Bc = \text{false}$ denotes a mismatch of Be and Bs .
- There is a random variable $S' \in \mathbf{S}'$, \mathbf{X}' with $\text{inView}(S') = v'$, where S' is a duplicate of the FBS-NT's Structure S with $\text{inPerspective}(S) = v$.

Factorization. Given a conditional dependency $P(X | Pa(X))$ in $P(\mathbf{X})$, let $A_{X \cup Pa(X)} = \{v_1, \dots, v_n\}$ be the set of mutually distinct Perspectives of variables $X \cup Pa(X)$, and let $A'_{X \cup Pa(X)} = \{(v'_1, \dots, v'_n) | \text{instanceOf}(v'_1) = v_1, \dots, \text{instanceOf}(v'_n) = v_n, v_i \in A_{X \cup Pa(X)}\}$ be the set of all possible corresponding View tuples (n-ary Cartesian product), there are duplicates $P(X' | Pa(X'))$ of the FBS-NT's $P(X | Pa(X))$ of the following types for every tuple (v'_1, \dots, v'_n) in $A'_{X \cup Pa(X)}$ that forms a path in G_{View} :

- $X' \in \mathbf{F}'$ and $Pa(X') \subseteq \mathbf{F}' \setminus X'$ (Function implicates Function)
- $X' \in \mathbf{Be}$ and $Pa(X') \subseteq \mathbf{F}'$ (Function expects Behavior)
- $X' \in \mathbf{Be}$ and $Pa(X') \subseteq \mathbf{Be} \setminus X'$ (Behavior implicates Behavior)
- $X' \in \mathbf{Bs}$ and $Pa(X') \subseteq \mathbf{Bs} \setminus X'$ (Behavior implicates Behavior)
- $X' \in \mathbf{Bs}$ and $Pa(X') \subseteq \mathbf{S}'$ (Structure exhibits Behavior)
- $X' \in \mathbf{S}'$ and $Pa(X') \subseteq \mathbf{S}' \setminus X'$ (Structure implicates Structure)

Table 1. Concepts of the FBS-NT

Type	Perspective	Name	States
Function	Office	Being Flexible	Important, Unimportant
	User	Being Efficient	Important, Unimportant
Behavior	Office	Adjustability	High, Low
		Enclosure	High, Low
	User	Distractions	High, Low
		Quiet Work	Plenty, Moderate
Structure	Office	Layout	Open-Plan Office, Cell Office
	Office	Partitions	Cubicle, Acoustic Curtain, Solid Walls

5 Application for Sales Call Support

Consider the following example of our domain of interest: Flexibility in dealing with office changes, e.g. staff churn, is a frequent requirement in office fit-out projects. “Organisations are constantly required to deal with change, so office facilities need to be designed to be flexible to adapt to future changes” [2]. An open-plan office layout may offer the required adaptability. But the type of office layout may also influence the occupant’s efficiency. Depending on their work type occupants need a distraction-free environment for doing concentrated quiet work. “An acceptable acoustic environment may be achieved in an open-plan setting for some of those behaviour patterns, but not all” [20]. An office designer may address this by providing a proper enclosure, such as solid walls or noise reducing curtains, to those workspaces that have high demands on acoustic privacy.

To formalize this knowledge in an FBS-NT we first define the problem/solution space, by providing a set of Function, Behavior and Structure variables and assign these to Perspectives as shown in Table 1. Building on the defined variables we connect the problem and solution parts with conditional probabilities. In the same manner as depicted in Table 2, we encode the following statements as probability tables: $P(B_1 | F_1)$ to be flexible with respect to future changes, an office should be highly adjustable; $P(B_3 | F_2)$ to work efficient, users should not be distracted; $P(B_1 | S_1)$ open-plan offices are highly adjustable, while cell offices are rather rigid; $P(B_2 | S_1, S_2)$ given an open-plan office, acoustic curtains provide a better enclosure than cubicles, and solid walls provide the best enclosure, but these are only available in cell offices; $P(B_3 | B_2, B_4)$ if a user group has a high amount of quiet work to do, but has not a sufficiently enclosed workspace, distractions will be high.

Now imagine a lead qualification situation where the lead requires the new office to accommodate two user groups with different needs in doing quiet work, e.g. a project management and a software engineering department. The instantiated FBS-BN is shown in Fig. 2. While users of the software engineering group spend most of their time with concentrated computer work, project managers are more concerned with communicative acts, like meetings and phone calls (cf. [20]). Given that both flexibility and efficiency are important goals for the lead,

Table 2. Encoding of “to be flexible with respect to future changes, an office should be highly adjustable” as probability table

Being Flexible F_1	Adjustability B_1	Probability $P(B_1 F_1)$
Important	High	1.0
Important	Low	0.0
Unimportant	High	0.5
Unimportant	Low	0.5

the preferable solution would be to have an open-plan office with acoustic curtains. Bayesian inference on the FBS-BN will exactly express this in terms of a higher probability for the state Acoustic Curtain of variable Partitions $P(S_2^{v'_1} = \text{Acoustic Curtain})$, if we set the probabilities to 1 for Being Flexible $P(F_1^{v'_1} = \text{Important})$, and Being Efficient $P(F_2^{v'_2} = \text{Important})$, $P(F_2^{v'_3} = \text{Important})$, and Quiet Work $P(Be_4^{v'_2} = \text{Moderate})$, $P(Be_4^{v'_3} = \text{Plenty})$, and 0 in all other cases. These probabilities represent answers given to the questionnaire.

To highlight those variables that are important in determining the problem/solution space but have not been answered yet, we use a scoring function based on the inferred probabilities. We define a measure of uncertainty $S[P(X)]$ as the Kullback-Leibler divergence [14] of $P(X)$ with respect to a discrete uniform distribution of the same size n and normalize it to $[-1, 0]$:

$$S[P(X)] \stackrel{\text{def}}{=} \left(\sum_x P(X=x) (\log P(X=x) - \log \frac{1}{n}) \right) / \log \frac{1}{n}$$

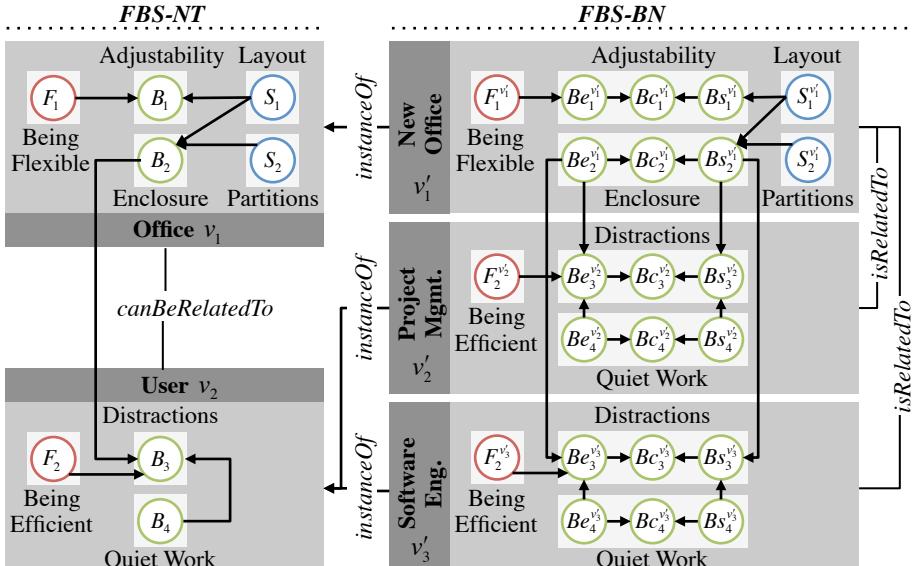


Fig. 2. Instantiation Example

This score is high (close to 0) if the inferred probabilities of X show a high ambiguity and are close to the uniform distribution, i.e. the variable's state is unknown and should be assessed by the salesperson. By rating all variables $X \in \mathbf{F}' \cup \mathbf{Be} \cup \mathbf{S}'$ with $S[P(X)]$ we can generate a ranked list of concepts. Questions that ask for these concepts are then presented to salesperson in form of a questionnaire, specifically highlighting higher rated concepts.

6 Conclusions and Future Work

We have presented a computational representation of the FBS framework, which specifically addresses the uncertainty inherent in the vendor's perceptions of a lead's demands. It is used to assist lead qualification by highlighting requirements and solution components that should come to speak in sales calls.

Currently we are integrating this representation in an prototype application, that resembles a questionnaire for sales call preparation. We expect this system to have a positive effect on a salesperson's performance in lead qualification situations, and look forward to test this assumption empirically.

Acknowledgement. This work was partially funded by the German Federal Ministry for Education and Research (BMBF, contract 17N0409). The authors would like to thank Sabine Janzen, Andreas Filler and Tobias Kowatsch for valuable discussions.

References

1. Aamodt, A., Langseth, H.: Integrating Bayesian Networks into Knowledge-Intensive CBR. In: Proc. of AAAI Workshop on CBR Integration, pp. 1–6 (1998)
2. Baccarini, D., Bateup, G.: Benefits management in office fit-out projects. Facilities 26(7/8), 310–320 (2008)
3. Bonnema, G.M., van Houten, F.J.A.M.: Use of models in conceptual design. J. Eng. Des. 17(6), 549–562 (2006)
4. Brinkmann, J.: Buying Center-Analyse auf der Basis von Vertriebsinformationen. Deutscher Universitäts-Verlag, Wiesbaden (2006)
5. Christophe, F., Bernard, A., Coatanéa, É.: RFBS: A model for knowledge representation of conceptual design. CIRP Annals - Manufacturing Technology 59(1), 155–158 (2010)
6. Cross, N.: Design Cognition: Results From Protocol And Other Empirical Studies Of Design Activity. In: Eastman, C., Newstatter, W., McCracken, M. (eds.) Design Knowing and Learning: Cognition in Design Education, pp. 79–103. Elsevier, Oxford (2001)
7. De Campos, L.M., Fernández-Luna, J.M., Huete, J.F.: Bayesian networks and information retrieval: an introduction to the special issue. Inform. Process. Manag. 40(5), 727–733 (2004)
8. El-Sawalhi, N., Eaton, D., Rustom, R.: Contractor pre-qualification model: State-of-the-art. Int. J. Proj. Manag. 25(5), 465–474 (2007)

9. Erdena, M.S., Komotoa, H., van Beeka, T.J., D'Amelioa, V., Echavarriaa, E., Tomiyama, T.: A review of function modeling: Approaches and applications. *AI EDAM* 22(2), 147–169 (2008)
10. Gero, J.S., Kannengiesser, U.: The situated function-behaviour-structure framework. *Des. Stud.* 25(4), 373–391 (2004)
11. Helms, M.E., Goel, A.K.: From Diagrams to Design: Overcoming Knowledge Acquisition Barriers for Case Based Design. In: Proc. of the Third Intl. Conf. on Design Computing and Cognition (DCC 2008), pp. 341–360 (2008)
12. Ingram, T.N., Schwepler Jr., C.H., Hutson, D.: Why Salespeople Fail. *Ind. Market. Manag.* 21(3), 225–230 (1992)
13. Jalkala, A., Cova, B., Salle, R., Salminen, R.T.: Changing project business orientations: Towards a new logic of project marketing. *Eur. Manag. J.* 28(2), 124–138 (2010)
14. Kullback, S., Leibler, R.A.: On Information and Sufficiency. *Ann. Math. Stat.* 22(1), 79–86 (1951)
15. Labrousse, M., Bernard, A.: FBS-PPRE, an Enterprise Knowledge Lifecycle Model. In: Bernard, A., Tichkiewitch, S. (eds.) *Methods and Tools for Effective Knowledge Life-Cycle-Management*, pp. 285–305. Springer, Berlin (2008)
16. Matthews, P.C.: Bayesian networks for engineering design decision support. In: Proc. of the Intl. Conf. of Data Mining and Knowledge Engineering (ICDMKE 2007), pp. 284–289 (2007)
17. Mohemad, R., Hamdan, A.R., Ali Othman, Z., Mohamad Noor, N.M.: Decision Support Systems (DSS) in Construction Tendering Processes. *IJCSI* 7(2), 35–45 (2010)
18. Moncrief, W.C., Marshall, G.W.: The evolution of the seven steps of selling. *Ind. Market. Manag.* 34(1), 13–22 (2005)
19. Nikolaidis, E., Mourelatos, Z.P., Pandey, V.: *Design Decisions under Uncertainty with Limited Information*. CRC Press/Balkema, Leiden (2011)
20. Olson, J.: Research about office workplace activities important to US businesses And how to support them. *J. Facil. Manag.* 1(1), 31–47 (2001)
21. Pearl, J.: *Causality: Models, Reasoning and Inference*. CUP, Cambridge (2000)
22. Pelham, A.: Sales Force Involvement in Product Design: The Influence on the Relationships Between Consulting-Oriented Sales Management Programs and Performance. *J. Market. Theor. Pract.* 14(1), 37–55 (2006)
23. Tang, A., Nicholson, A., Jin, Y., Han, J.: Using Bayesian belief networks for change impact analysis in architecture design. *J. Syst. Software* 80(1), 127–148 (2007)
24. Tuli, K.R., Kohli, A.K., Bharadwaj, S.G.: Rethinking Customer Solutions: From Product Bundles to Relational Processes. *J. Market.* 71(3), 1–17 (2007)
25. Van der Haar, J.W., Kemp, R.G.M., Omta, O.: Creating Value that Cannot Be Copied. *Ind. Market. Manag.* 30(8), 627–636 (2001)
26. Wang, G., Netemeyer, R.G.: Salesperson creative performance: conceptualization, measurement, and nomological validity. *J. Bus. Res.* 57(8), 805–812 (2004)
27. Xue, D., Yang, H.: A concurrent engineering-oriented design database representation model. *Comput. Aided Des.* 36(10), 947–965 (2004)
28. Yoshioka, M., Umeda, Y., Takeda, H., Shimomura, Y., Nomaguchi, Y., Tomiyama, T.: Physical concept ontology for the knowledge intensive engineering framework. *Adv. Eng. Informat.* 18(2), 95–113 (2004)
29. Zha, X.F.: Artificial Intelligence and Integrated Intelligent Systems in Product Design and Development. In: Leondes, C.T. (ed.) *Intelligent Knowledge-Based Systems*, pp. 1067–1123 (2005)

Generic and Specific Object Recognition for Semantic Retrieval of Images

Martin Klinkigt¹, Koichi Kise¹, and Andreas Dengel²

¹ Graduate School of Engineering, Osaka Prefecture University

² German Research Center for Artificial Intelligence

Abstract. Since the availability of large digital image collections the need for a proper management of them raises. New technologies as annotations or tagging support the user by doing this task. However, this task is time-consuming and, therefore, automatic annotation systems are requested. Working outside of controlled laboratory environments this request is challenging. In this paper we propose a system automatically adapted to the user's needs, providing useful annotations. We utilize Wikipedia to learn instances and abstract classes. With an evaluation in a complex use-case and dataset we show the possibility of such an attempt and achieve practical recognition rates of 26% on specific instance and 64% on abstract class level.

Keywords: SIFT, shape model, SVM, image management, specific object recognition, generic object recognition.

1 Introduction

The request for semantic management of images raises, since the the number of digital images increases rapidly. With such semantic management approaches the content, topics, events etc. of images can be expressed in a multi-dimensional way. This is much superior as compared with classic folder structures available on desktop computers. By naming these semantic concepts the concrete images can be found easily. The burden of this solution is that the user has to provide all this information by hand. For text documents automatic tools like iDocument [1] are available. By analysing the full text content, this system performs an entity recognition to close the gap between low-level content and high-level semantic concepts.

Text documents are well researched concerning the automatic generation of meta-information. Working with images this aspect is rather complex. On the market available systems like iPhoto, Picasa or Flickr introduce image management with the help of tags. These systems also introduce face recognition to categorize images showing a certain person. However their recognition performance is often low resulting in much hand work. Additionally, for more generic objects, for example cars, the tags must be provided by the user manually which is time-consuming when facing thousands of images.

To acquire semantic information for images, image processing and especially object recognition is applied. Although humans are exceptional in recognizing an object, computers can simply compare pixel by pixel. If two images are showing the same content but under different resolutions, the pixel information is not identical.

Therefore, it is common to introduce higher level features which abstract from the pixel information. SIFT features [6] have been proposed and remarkable results were achieved concerning various problems, such as changes in the resolution or lighting conditions. SIFT features are a powerful means to match information between images. However, no human understandable information can be inferred from their mathematical values.

In this paper we present a system providing semantic concepts automatically closing the semantic gap between images and their content description. We propose the use of adapted recognition systems on instance level and flexible support vector machines on generic level. Our system utilizes Wikipedia and a Semantic Desktop to link the images with the concepts of the user and keeping the additional work low. In an evaluation on a difficult dataset our proposed system still achieves meaningful recognition rates of 29% on instance level and 64% on more abstract levels.

2 Related Work

Datta et al. survey in their article [3] almost 300 contributions in the field of content-based image retrieval. Most of these systems working purely deriving high-level semantics, e.g. events, or persons from plain image information. However, the information such systems can utilize is limited to the ability to organize the image data [10].

Sandhaus et al. discovered in [9] a new trend in recent years. With the availability of social networks with image sharing like Flickr, Facebook or LableMe large annotated image collections are available. By using the infomration how images are already integrated and used, high-level meaningful semantic annotations can be generated.

Overall the use of ontology for image management is quite frequently applied from many different viewpoints. Kong et al. proposed such a system for the private image collection [5]. However, the user has to integrate all his images by hand into the ontology which makes such approaches a time-consuming task. One step in direction of automatic annotation of images was done by Yang et al. [11]. With the help of image processing strategies they are able to arrange a set of images automatically. However, still one link is missing. The classifier must be pre-learned by hand provided training images.

Learning such classifiers in an automatic way was considered by Renn et al. [7]. They utilized the provided images and annotation from Flickr for a tagging system and compared the performance with a professionally annotated image set. Their results from such community-driven image databases were not satisfying, since the annotations are subjective or non-relevant. Flickr was also utilized by

Rohrbach et al. [8]. The objective of Rohrbach et al. is to use these annotations to recognize new, unseen objects for which no training image is provided. Rohrbach et al. did not present these annotations to the user, leaving the question about their quality and relevance. Additionally, the authors simultaneously use six different feature descriptors making it difficult to compare their approach with other systems.

In our system we integrate different approaches to provide helpful annotations. As in the spirit of Renn et al. the system also uses automatically mined annotations and trains the classifier based on the images downloaded from the Internet. However, we do not utilize Flickr, since the quality of the provided labels is not high enough. We utilized Wikipedia and DBpedia which create a well organized ontology. We embed our recognition system into a Semantic Desktop, giving us the ability to utilize social aspects as discovered by Sandhaus et al. However, the main focus of this paper is to address the challenging problems by working in community driven image collections and annotations. The novelty lies within the object recognition especially adapted for this complex and challenging task.

3 Overview

With this section we setup the vocabulary for this paper. We start with RDF in section 3.1 which we use to represent knowledge and then we discuss about different types of knowledge in section 3.2.

3.1 Resource Description Framework

The Resource Description Framework (RDF) is an international standard of the W3C and has become the base of the semantic web. With the help of RDF statements information about entities can be expressed. Such a statement can be seen as a simple sentence in natural language with subject, predicate and object. As example we can give: KES 2011 *is a* conference or KES 2011 *is held in* Kaiserslautern.

3.2 Knowledge in Image Processing

In recent research the use of knowledge in image processing becomes more and more attractive. Knowledge is often applied to increase the recognition performance or to enable new use-cases. This semantic information varies from simple tag information to related parts-models describing the structure of an object. For all these systems the semantic information is modelled by hand and used for recognition purposes. However, these systems are often build for a certain interest and context (recognition of baseball players, sailing ship on the ocean, etc.). Generic applications of such systems are hardly possible. Our utilization of knowledge is different and, therefore, we separate between *inside* and *contextual* knowledge.

Inside Knowledge. As inside knowledge we categorize all information used in the system internally. This information is used only by the system for recognition purposes and can become as technical as the system requires and not intended to be seen or provided by the user. This can be annotation data for images as for example used in [8] or part relation models as proposed in [12]. Characteristic for this knowledge is that it has no use outside of the system to present it to the user nor can it be provided by the user.

Contextual Knowledge. A different way of utilizing semantic information is to communicate with the system and provide information for the training and recognition process dynamically. To give an example: It is hard to name a certain guitar in an image without knowing something about its context. However, with the knowledge that the images was taken during Ben's birthday it becomes simple to name the guitar as "Ben's guitar". In that example even a human would have the same problems as a computer. As natural it sounds, the problem would be how to acquire such contextual knowledge. The user could provide such information directly. The result would be that the user could also directly name the guitar rather than telling the system about Ben's birthday. The use of this contextual knowledge is the motivation of top-down image processing approaches as explained with the next sections.

4 Proposed Method

The major concern of systems supporting a non-professional user with the management of image data is time. The user wants to get a time benefit and is not willing to train a system over long time, nor he is patient enough to wait several hours until the system has finished its work. If the system needs more time to setup as the user would need to annotate the images by hand, there is no reason to use such a system.

This does not effect the processing time of the system, since computers become faster every day. The problem is the training of the system to recognize the concepts of the user's interest. Available annotated images are normally not well prepared for object recognition which means, the object is not segmented from the background. Existing systems often rely on this fact.

We avoid this problem of the training by utilizing Wikipedia. The user has only to name a category of interest, e.g., "Sightseeing in Japan" and the system access Wikipedia and learn its classifiers from the provided images. In the following sections we discuss the core problem in object recognition and present our solution.

4.1 Problem for Object Recognition

By working on such unprepared image data not especially prepared for object recognition purposes, erroneous matching of information is the core problem. This means different objects have a high similarity to each other. This is the case, if information provided during training is characteristic for different classes one

wants to distinguish among. This is often the result, if the images are not well segmented and contain background. Objects appearing in the background are often shared between different classes of objects. By taking images of sightseeing spots it will immediately become clear. An image of the Eiffel Tower might contain persons somewhere. For other spots it could be a tree.

If the information of the background overweight the information from the object of interest, no reliable recognition can be performed. One has to find a way to decide the images into parts belonging to the object or the background. One approach could be to take many images of the same object and extract the common information. This relies on the condition that also the background changes in the images. From our given example it is obvious that this can not be guaranteed. Second, in our purpose of learning in an automatic way, we can not be sure to always have many images. In the case of only one image of the object, the separation from the object must be done in a more sophisticated way.

4.2 Proposed Solution on Instance Level

We address the above stated problem by observing that:

Information from the background of different images matches in an *unstructured* manner.

Unstructured means that even if background objects are shared among different images, their relative location to each other and the object of interest are different. Cars do not appear always at the same position, trees do not always have the same structure.

The benefit of this observation is that one training image is already enough to address the problem of background clutter. Suppose we have matching information, in our case local features, the question would be how to identify whether a certain match is structured or not. We construct a model of the local arrangement of features to express these structures. Defining a model is not straightforward. One could think of constructing a graph defining the relative position of the features to each other as expressed in Fig. 1(a) with solid lines. In such a model features of the object and the background are part of the same graph. A verification of the structure becomes hardly possible, since the background features are unstable in their position.

For the details on the implementation we have to refer to the work [4], since in this paper we can only summarize our model. To solve the problem with one approach alone is difficult. Ridged object can be described based on its details or its global appearance. Working with details the exception concerning correctness is high; the eyes are not at the place of the mouth. Considering the global shape this exception is lower; the windows and doors of a house can be at different locations. Therefore, in a local region the model should be stricter and in a global view more flexible. Figure 1(b) gives a picture of this consideration. Solid lines again visualize hard relative positions between features. A small agglomeration of such features forms something like an island. Indicated with dashed lines are flexible bridges (connections) between these islands.

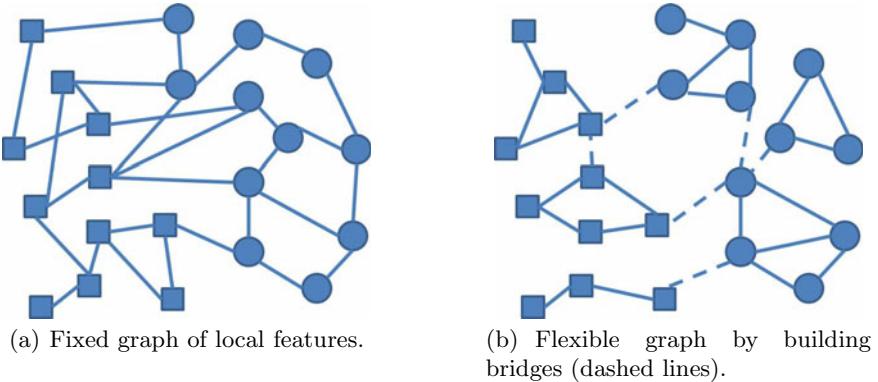


Fig. 1. Comparison between global strict model and global flexible and local strict model. Squares indicate features from the object, circles from the background. Used relative positions between local features are shown by solid lines. Dashed lines show relative positions of graphs.

For the local shape we utilize a shape context [2] consisting of a log-polar histogram, segmenting the surrounding area in clear defined sectors. To build the bridges we work with a common reference point. Concerning this point the position of the local parts are expressed. In this model the influence of background is limited to the islands.

4.3 Proposed Solution on Class Level

The above presented solution is considerable on instance level or in other terms on specific object level. However, if the concrete instance is not a part of the database, a correct result can hardly be returned. However, we still want to present the user meaningful annotations. In former solutions we utilized the class hierarchy of Wikipedia for this task. By browsing from the learned instance up in the categorization we could present the user annotations on abstract level. However, this is only possible for one or two levels, since the categories become soon meaningless, as for example *place*, *person* or *Religious organizations established in the 1320s*.

Multiple Categorization. Annotation on the base of instances, as on specific object level, has no needs for any discussion. Working with abstract classes it is worth to discuss meaningful annotations. One has to consider that different users have different views and opinions about which annotations are proper and which not. If the user has a certain interest in cars, he will model his knowledge very detailed by providing many classes as companies (Toyota, BMW, etc.), types of cars (van, sports car, truck, etc.). This results in a very detailed and professional view leaving general annotations behind as for example *car*.

If the user does not have such an interest in cars, since he is not such a professional, he will not understand such detailed annotations. He would be satisfied with

one simple annotation *car*. However, since this information is obvious and is not provided by the first user, the non-professional user will not find any “car” related information. To address this problem, the annotations should not only be provided in the form of keywords. The annotations should contain information about themselves and be organized in a structure ranging from detailed to abstract.

Letting one user model his knowledge about certain aspects will always miss some information. Also different users have different point of views. Therefore, the system must be able to handle multiple views or categorizations for the objects. The difficult question is, how to collect this knowledge in advance to present it to the user. We utilize for this task Wikipedia. An object is categorized in a general structure, for example, building or car. Furthermore, users provide more detailed information about the objects, as related persons, epoch, history, etc. Our system works autonomously in collecting all this data from Wikipedia without any human made selection. Limitations are only set by memory and calculation capacity to handle millions of images and annotations.

Other information sources are difficult to utilize. Databases like Imagenet are only keyword based and, therefore, important information is missing. Equivocal objects as for example “head” will bring the head of animals, hard-disks, tools, nails, a company etc. Arranging these different type of objects to form one classifier will not lead to meaningful results. The classifier is more randomly guessing whether an image is showing a “head” or not. A selection of proper annotations and their examples would be needed and has to be done by a human. Additional, only one hierarchy is provided for the categorization of a certain object. As discusses in the previous paragraphs, this is not sufficient enough to provide useful annotations for all users.

In this paper we present the first step by working on Wikipedia with a selected number of classes, to keep the problem solvable with the available scientific environment.

Implementation Details. In this paper we utilize a support vector machine (SVM) on abstract levels. The system is using the information available from Wikipedia to construct classes on abstract level. The instances are grouped by their types and topics. The system re-use the same local features as in the previous section. These features are cast to their nearest visual word which results in a frequency histogram for each image. In experimental evaluations we observed that a number of 2000 visual words leads to practical results. With a larger number of visual words, the trained SVM becomes over-specific, since the histograms are too sparse.

With a Gaussian kernel we are not able to achieve reliable results. The images as provided by Wikipedia are too various. Also the number of images available per category is quite limited. With around ten images per instance and normally not more than ten instances per class, the system has to train the classifier with only around 100 images. Keeping in mind the high variance of the images so is this number very low. In our preliminary done experiments, a χ^2 -kernel is most promising. Via grid search on the typical parameter setting we found the kernel bandwidth of 40 is practical for this kernel.



Fig. 2. Example query images from the temple data set. From left to right two images of Kinkaku-ji, two images of Ginkaku-ji and two images of Kiyomizu-dera are shown.

Table 1. Results for the temple dataset. Shown is the mean average precision (mAP) in percentage. SIFT utilize only PCA-SIFT features, WGC refers to the weak geometric consistency, PM our former proposed method for instance level and SVM for class level.

	instance			class SVM
	SIFT	WGC	PM	
Ginkaku-ji	17.13	15.23	25.39	69.05
Kinkaku-ji	23.13	26.04	29.05	47.77
Kiyomizu-dera	18.92	24.37	23.21	76.19
overall	19.72	21.87	25.81	64.34

5 Experiments

Publicly available datasets are not suited to evaluate a system with our motivation by learning from public databases as Wikipedia. Available datasets have a certain emphasis on a certain interest in the images. This results in limited classes, e.g., only detecting horses, or images for learning and testing looking quite similar.

Therefore, we prepared our own database by utilizing Wikipedia as training base. For this task we downloaded all images provided for the categories “world heritage sites in japan” and “national treasures in japan”. From the 1680 images provided for 126 objects in the category the system creates a database. Working on this database, the objects look quite similar, since they were mainly temples and shrines. Even humans have problems correctly naming the shown objects or class. All images of a certain instance are used to train the specific object classifier as explained in section 4.2. For the generic level the system analyses the rdf *type* property. All instances belonging to a certain type were grouped to one class, e.g., temple or castle.

In our evaluation the query dataset consists of images from temples and shrines. These images were not taken for the propose of object recognition and simulate the behavior of an tourist. The objects are shown with the surroundings and people in front of them. Figure 2 shows an example sequence of these images.

The results are shown in Table 1 split for the different objects. SIFT refers to a system only utilizing local features. In this approach erroneous matching is the major problem. The weak geometric consistency (WGC) proposed by Jegou et al. has achieved only a minor improved overall performance. The shape information utilized in the WGC can address the problems of erroneous matching in a limited

way. From this results on instance level we can conclude that in this complex recognition problem our proposed method performs better than existing systems. However, with 30% the recognition is relative low. Keeping in mind that the system is trained automatically on images as provided by Wikipedia, without any interaction of the user, the system still support him with his image annotation.

From the generic class recognition we can draw the main conclusion. Even if the scope of the experiment is limited, the results look promising which leads to the hypothesis that the classes organized by Wikipedia are of high quality. So even if the system fails to recognize the concrete instance (not a part of the database or confusion with different objects), the possibility is high to return the correct class. Under such conditions, were the systems rely only on a community for training, we could provide a proof-of-concept with this paper. It is possible to achieve meaningful recognition rates and annotations by collecting data from Wikipedia without any filtering done by human.

6 Conclusion

In this paper we developed a prototype aiming to support non-professional users managing their image collections. We integrated image processing technologies working on instance as well as on class level to recognize objects and propose annotations. For this task the system steps into online information sources, namely DBpedia and Wikipedia and learns categories depending on the user's interest. Compared to existing systems we train the classifier completely based on the images from Wikipedia, rather than using them to extend well prepared training sets. To our knowledge we are also the first providing the concepts from Wikipedia as annotations to the user. Former systems used them internally for recognition purposes. The possibility to perform all required steps setting up the recognition system automatically is the major advantage of our proposed system. With an evaluation on a difficult dataset we have shown that in such a difficult use-case, practicable performance of 26% on specific instance and 64% on abstract class level can be achieved.

Future work will focus on an extensive evaluation and deeper analysis of the knowledge available from Wikipedia. Furthermore, the integration of a top-down approach in object recognition will be considered. The needed data for this task is provided by the Semantic Desktop in which we integrated our recognition system.

Acknowledgments. This work was supported in part by the Grant-in-Aid for Scientific Research (B) (20300049) from Japan Society for the Promotion of Science (JSPS).

References

1. Adrian, B., Hees, J., van Elst, L., Dengel, A.: idocument: Using ontologies for extracting and annotating information from unstructured text. In: Mertsching, B., Hund, M., Aziz, Z. (eds.) KI 2009. LNCS (LNAI), vol. 5803, pp. 249–256. Springer, Heidelberg (2009)

2. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. *IEEE PAMI* 24(4), 509–522 (2002)
3. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.* 40, 5:1–5:60 (2008)
4. Klinkigt, M., Kise, K.: From local features to global shape constraints: Heterogeneous matching scheme for recognizing objects under serious background clutter. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) *ACCV 2010, Part IV. LNCS*, vol. 6495, pp. 64–75. Springer, Heidelberg (2011)
5. Kong, H., Hwang, M., Kim, P.: Pims(personalized image management system) using ontologies. In: The 7th Int. Conference on Advanced Communication Technology, ICACT 2005, vol. 1, pp. 277–280 (2005)
6. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proc. of ICCV, p. 1150 (1999)
7. Renn, M., van Beusekom, J., Keysers, D., Breuel, T.: Automatic image tagging using community-driven online image databases. In: Detyniecki, M., Leiner, U., Nürnberg, A. (eds.) *AMR 2008. LNCS*, vol. 5811, pp. 112–126. Springer, Heidelberg (2010)
8. Rohrbach, M., Stark, M., Szarvas, G., Gurevych, I., Schiele, B.: What helps where - and why? semantic relatedness for knowledge transfer. In: CVPR (2010)
9. Sandhaus, P., Boll, S.: Semantic analysis and retrieval in personal and social photo collections. *Multimedia Tools and Applications* 51, 5–33 (2011)
10. Sawant, N., Li, J., Wang, J.: Automatic image semantic interpretation using social action and tagging data. *Multimedia Tools and Applications* 51, 213–246 (2011)
11. Yang, J., Fan, J., Hubball, D., Gao, Y., Luo, H., Ribarsky, W.: Semantic image browser: Bridging information visualization with automated intelligent image analysis. In: Proc. IEEE Symposium on Visual Analytics Science and Technology (2006)
12. Yao, B., Fei-Fei, L.: Modeling mutual context of object and human pose in human-object interaction activities. In: IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, USA (June 2010)

A Lane Detection and Tracking Method for Driver Assistance System

Nadra Ben Romdhane¹, Mohamed Hammami², and Hanene Ben-Abdallah¹

¹ MIRACL-FSEG, Sfax University, Rte Aeroport Km 4, 3018 Sfax, Tunisia

nadrabenromdhane@yahoo.fr

hanene.benabdallah@fsegs.rnu.tn

² MIRACL-FS, Sfax University, Rte Sokra Km 3 BP 802, 3018 Sfax, Tunisia

mohamed.hammami@fss.rnu.tn

Abstract. In driver assistance systems, lane detection and tracking are very crucial treatments to locate the vehicle and to track its position on the road. The aim of this study is to propose lane detection and tracking method. The first step in this method detects road limits on the first acquired image. The detected limits would be the input for the second step, namely the “tracking step”, which consists in providing a continuous detection of the limits in all frames by updating the previously detected limits. Lane departure is also analyzed for the lateral control of the vehicle. The approach presented here was tested on video sequences filmed by the authors on Tunisian roads, on a video sequence provided by Daimler AG as well as on the PETS2001 dataset provided by the Essex University.

Keywords: Lane detection, Lane tracking, kalman filter, Lane departure.

1 Introduction

Accidents caused by involuntary lane departure continue to represent an important part of accident-prone traffic conditions. Seeing the importance of their application, many driver assistance systems (mainly in the US, Japan and Europe) have been developed [1-2]. Their main objective is to assist the driver by discharging him/her from some tasks. An important component of these systems is the analysis of image sequences recorded with cameras mounted on the moving vehicle. These image sequences are analyzed in order to support the driver in real traffic situations.

To avoid accident-prone situations, our suggested lane departure warning solution is based on two steps: lane detection and tracking. Since the majority of roads have lane markings on both sides, we are interested in roads where markings are presented in the form of two parallel linear or curved ribbons, with continuous or dashed lines.

This paper is organized as follows: Section 2 describes the different steps of typical lane detection and tracking process. Section 3 details our proposed method. Section 4 presents a set of experimental evaluation and comparison results. Section 5 concludes by summarizing the major contributions of the present work.

2 Lane Detection and Tracking Process

The aim of lane detection and tracking based on image processing is, first, to locate in the first sequence frame the Lane Limits (*LL*) of the road in which the vehicle is engaged and then, to track these limits in the remaining frames.

2.1 Lane Detection

Various lane detection methods have been proposed in the literature (*cf.*, Table 1). These methods can be broadly grouped in two approaches, namely the model-based approach [3-8] and the feature-based approach [9-13].

Table 1. Lane detection methods

	Method	Description	Technique	References
Model based approach	Parametric models	Represent the lane boundaries using few parameters. They are robust with the noises but only the lanes whose shape is described in the model can be detected.	Hough transform linear functions	[3,4] [5, 6]
	Explicit models	Detect different shapes of road. However, the segmentation result depends enormously on the initialization of the model, they present great sensitivities to noises and require considerable execution times.	B-snake	[7, 8]
Feature based approach	Rule-based methods	Search the forms similar to those of lane marking limits according to user-defined classification rules.	user-defined classification rules	[9,10]
	Supervised training methods	Produce the classification rules automatically based on a training dataset images to search the forms similar to those of lane marking limits.	decision tree classifier multi-layer perceptron neural network	[11] [12]
	Tracking methods	Consider iteratively, from a starting point, different segments from which the segment which corresponds best to the required structure is kept.	Markov process	[13]

With the feature-based approach, the road is assumed to have a very clear marking with contrasted edges in order to obtain good results, which is not usually the case. Moreover, this type of approach is sensitive to noise and partial coverage of markings by obstacles. On the other hand, a model-based approach is more robust against these problems. With this type of approach, lane limits can be modelled using parametric or explicit models.

2.2 Lane Tracking

The lane tracking step is required to minimize the noises, to minimize the execution time and also to predict the position of the limits in the following frames.

Different methods have been proposed to perform this tracking. These methods can be classified into two approaches, namely the deterministic approach [14-17] and the stochastic approach [13, 18].

Deterministic approach. The principle of the deterministic approach is that, under an initial state in the current frame, a unique possible future state will corresponds to each following frame. This approach includes different methods, such as filtering based methods [14, 15] and active contour based methods [16, 17]. These methods search for the nearest model to the extracted characteristics of the image while optimizing a performance measure. For the filtering based methods, tracking with Kalman filter [14] and Extended Kalman filter [15] are the most used techniques. For active contour based methods we can denote the use of the Bspline [16] and the gradient vector flow [17].

Stochastic approach. The stochastic approach is adopted for lane tracking when there are uncertainties in the observations due to two sources: the noise of the cameras, and/or incomplete information. In these cases, a probability function tracks the *LL* more correctly. This approach includes various methods, such as the Observable Markov Model based methods [13], the Hidden Markov Model based methods [18].

The tracking step allows controlling the lateral navigation of the vehicle through a lane departure warning module. This module is an important part in a driver assistance system; it controls the lateral navigation of the vehicle and alerts the driver when he begins to move out of his lane without activating the turn signals. In this context, different systems were proposed. These systems can be classified in two types: lane departure warning systems (LDWS) and lane keeping systems (LKS). The first type warns the driver if the vehicle is leaving its lane using visual, audible, and/or vibration warnings. The second type warns the driver and if no action is taken, then the system automatically takes steps to keep the vehicle in its lane. Citroën¹ was the first in Europe to offer such a module on their 2005 C4, C5 and C6 models. This system uses infrared sensors to monitor lane markings on the road surface. A vibration mechanism in the seat alerts the driver of deviations. In 2009, Mercedes-Benz¹ offered a lane keeping assistance function on the new E-class that warns the driver with a vibrating steering wheel if the vehicle starts to leave its lane. In 2010, Kia Motors¹ has offered the 2011 Cadenza premium sedan with an optional lane departure warning system. The system works by using an optical sensor on both sides of the car. It uses a flashing dashboard telltale and emits a warning when a lane marking is being crossed. The system used by General Motors in 2007 uses the core technology from Mobileye² based on camera images analyses that detect lane boundaries, measures the position of the vehicle relative to lanes, and provide indications of unintentional deviations from the roadway. All these systems are canceled when a turn signal is operating.

¹ <http://media.daimler.com>

² <http://www.mobileye.com>

3 Adopted Method for Lane Detection and Tracking

In our previous works [19, 20], we treated the lane detection step. Our algorithm is based on Hough transform and linear parabolic fitting. Based on this algorithm, we obtained good results in normal road conditions. Its effectiveness decreases in complex circumstances. This section extends this work by improving the lane detection results and by incorporating the tracking step allowing us controlling the lateral position of the vehicle through a lane departure warning step. Our proposed lane detection and tracking process is illustrated in Figure 1.

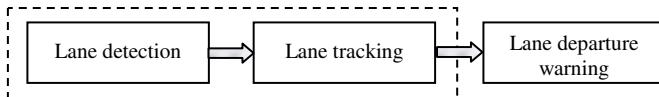


Fig. 1. Lane detection and tracking process

In the lane detection step, we detect the effective limits on the first acquired image. In the lane tracking step, we update in every new acquired image, the last detected limits to have a continuous detection within all the video sequence. The tracked limits allow us controlling the lateral navigation of the vehicle carrying the camera based on the lane departure warning step.

3.1 Lane Detection

Similar to the typical process, our proposed lane detection method adopts three steps: pre-processing, extraction of approximate pixels and lane detection steps.

In the pre-processing step, we apply a list of transformations to delimit the region of interest and to improve the contrast of the image. We delimit two rectangular regions to detect the linear portions of the road: ROI_r to detect the right limit, and ROI_l to detect the left limit (*cf.*, Figure 2.a). The improvement of the contrast is done by firstly applying the top hat transformation to extract the clear regions regardless of background variations (*cf.*, Figure 2.b) and secondly by enhancing the contrast by determining either the square or the cubic of the intensities according to the mean value of the pixels of the original image (*cf.*, Figure 2.c). For the extraction of approximate pixels step, we segment the image to extract the approximate regions of lane limit's markings, and then we select their edge pixels (*cf.*, Figure 2.d). In the lane detection step, we apply the Hough transform considering its rapidity and its robustness in the presence of noises and partial coverage of markings. In Hough transform, each line is a vector of parametric coordinates that are the orientation θ and the intercept to the origin P . Among the lines provided by this technique, we retained the segment with the maximum length in each region (*cf.*, Figure 2.e). Finally, we project the two retained limits up to the upper and lower sides of ROI_r and ROI_l (*cf.*, Figure 2.f).

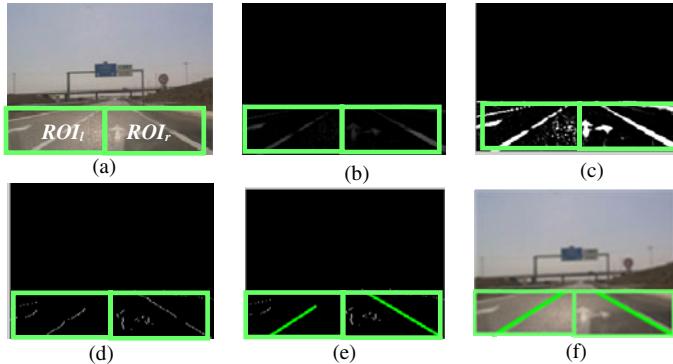


Fig. 2. Lane detection step

3.2 Lane Tracking

We track the detected limits in the near region to have a continuous detection of the limits during all the frames and then to control the lateral trajectory of the vehicle. We don't take into account the tracking in the far region because the curvature measurement can be unreliable especially during rain, fog or occlusions by obstacles. Thus, the stability of the tracking method can never be guaranteed; which is not allowed for the tracking of the vehicle trajectory in such a driver assistance system. We apply the kalman filter to perform the tracking.

The Kalman filter is an estimator with a linear model, which allows estimating, recursively and at each moment, the state vector X_t , of a dynamic system characterized by a process and a measurement equations. These equations are:

$$X_{t+1} = F X_t + w_t \quad (1)$$

$$Z_t = H X_t + v_t \quad (2)$$

where F is the transition matrix taking the state vector X_t , from time t to time $t+1$. The process noise w_t is assumed to be Gaussian, with zero mean and with covariance matrix Q . Z_t is the observable at time t and H is the observation matrix. The measurement noise v_t is assumed to be Gaussian, with zero mean and covariance matrix R .

As a result of the lane detection step, each lane limit l in the near region is characterized by two parameters that are the orientation θ and the intercept to the origin P . In a new acquired image, the lane limit position will have a low variation compared to the previous image with the change rates ($\Delta\theta, \Delta P$). Therefore, we use two kalman filters in order to estimate and update these parameters for the right and the left limits separately. We take as state vector $X_t = [\theta \ P \ \Delta\theta \ \Delta P]^T$, T returns the transpose of X_t , the observation matrix $H = [\theta \ P]$ is the measure of these parameters from the image.

3.3 Lane Departure Warning

Based on the tracked limits, we control the lateral trajectory of the vehicle based on two indicators to know the steering angle and the lateral offset [1, 2]. The first indicator corresponds to the angle formed between the longitudinal axis of the vehicle and the lane. It is defined by:

$$\text{Steering Angle: } \delta = |\theta_1 + \theta_2 - 180| \quad (3)$$

where θ_1 and θ_2 correspond to the orientations of the right and the left lane limits.

The second indicator corresponds to the lateral offset Δd of the vehicle compared to the middle of the lane. It can be obtained based on the vertical position h of the vanishing point:

$$\Delta d = h \times \tan(\delta) \quad (4)$$

4 Experimental Results and Performance Evaluation

In order to evaluate the performance of our proposed method, we carried out a series of experiments on various sequences. The test dataset is composed by several video sequences divided in three sets. The first set ($S1$) contains four video sequences which we had captured in Tunisian roads. The second set ($S2$) contains a night vision stereo sequence provided by Daimler AG³ (Germany). The third set ($S3$) contains the video sequences of the PETSc2001⁴ dataset provided by the Essex University (England).

In order to clarify the experimental conditions, the following sections will start by providing a brief overview of the test dataset and then proceed to present the experimental results and performance evaluation.

4.1 Evaluation of Lane Detection Step

In order to evaluate the performance of our method, we have implemented, for comparison, three proposed lane detection algorithms of the most known methods, including the Canny/Hough Estimation of Vanishing Points ‘CHEVP’ algorithm proposed by wang [7] and adopted by Tian [8] that is based on a local parametric detection method (Method A), the global parametric detection method (Method B) based on the model approach [6] and the tracking method (Method C) based on the feature approach [13]. We also present the detection results based on our previous work [19]. Table 2 illustrates the detection results of the different methods in different environment conditions.

The robustness of our method in comparison with our previous work and the three others is seen in frames presenting a combination of strong shadow, too spaced blurred lane markings, illumination variation and intense lighting (*cf.*, Figures a, c and g) due to the efficiency of our pre-processing step. Our method gives, as for the Method C, good results in the presence of a heavy rain (*cf.*, Figure c). Furthermore, it

³ <http://www.mi.auckland.ac.nz>

⁴ <ftp://ftp.cs.rdg.ac.uk>

detects efficiently the limits in case of curved lanes (*cf.*, Figures b and e) and in the presence of obstacles (*cf.*, Figure f). There remain few cases where the detection fails with our method and the others such as the presence of sharp curved lane, non flat road, congested road and also in case of intense raining as illustrated in Figure (h).

Table 2. Comparative evaluation of the lane detection step

	Environment condition	Original Image	Proposed Method	previous work	Method A	Method B	Method C
a	Strong bridge shadow + objects reflection						
b	Cloudy day						
c	Heavy Raining						
d	Night time						
e	Normal condition						
f	Obstacles						
g	Strong trees shadow						
h	Intense raining						

In order to further evaluate our lane detection method, we compared its detection performance with our previous work and with the three methods in terms of Recall, Precision and F-measure. We performed this evaluation on the set S3 since it is a known dataset and to provide our lane detection rates for comparison with other researches. This set is formed by a sequence (Video1) composed of 2866 frames and a sequence (Video2) composed of 2867 frames. We have manually detected the limits of 1000 frames: the first 500 frames of Video1 and the first 500 frames of Video2. The performance measure of the different methods is illustrated in Figure 3.

Based on our proposed method, we obtained an average rate of 92.04% for the Recall, 91.53% for the Precision and 91.76% for the F-measure. Our results are almost equal to the Method C. However, this method is characterized by its sensitivity to noises which lead it to present more cases of false detections than our method. On the

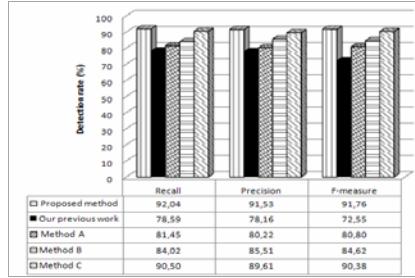


Fig. 3. Comparative performance evaluation of the five methods

other hand, our previous work, Method A and Method B give lower rates especially because of the presence of strong shadows and intense lighting in some frames.

4.2 Evaluation of Lane Tracking Step

To evaluate the lane tracking step, we compared the proposed method with the observable Markov model method (Method OMM) proposed by [13] that is based on the stochastic approach. To illustrate the tracking performance, we compared the obtained θ and P values of the right limit, based on our method and the Method OMM, to the ones we calculated on the reference lane limits for the first 500 frames of *Video2* (cf., Figure 4). In these figures, the thick curves illustrate the evolution the parameters according to the reference detected limits. The thin curves represent the results of our tracking method, and the dashed curves represent the results of the Method OMM.

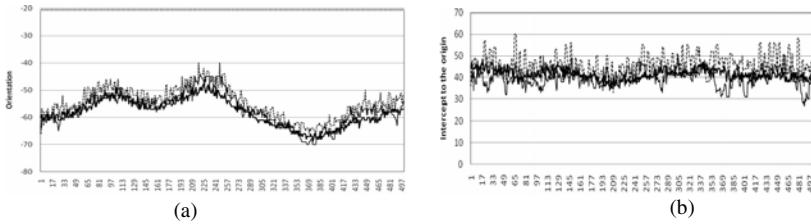


Fig. 4. Evolution of θ (a) and P (b) parameters of the right limit (*Video 2*)

In order to evaluate our lane tracking method, we compared the noise variance (Var) between the values obtained by each method and the ground truth. It is defined by:

$$y_{noise} = y_{method} - y_{groundTruth} \quad (5)$$

$$Var(y) = std(y_{noise})^2 \quad (6)$$

where y corresponds to θ or P values.

According to Figure 4.a, we obtained the variances 5.38 and 5.32 for θ parameter based on our proposed method and the Method OMM respectively. Similarly, according to Figure 4.b, we obtained the variances 21.41 and 27.57 for P parameter based on our proposed method and the Method OMM. Based on the obtained results, we can notice that the tracking with the Kalman filter provides a better prediction of the parameters than the Method OMM and it allows a good tracking of the lane limits.

4.3 Evaluation of the Lane Departure Warning Step

By tracking the lane limits, we control the lateral trajectory of the vehicle in order to prevent the driver from potential lane departure. To evaluate this control, we illustrate the evolution of the steering angle (cf. Figure 5.a) and the lateral offset (cf. Figure 5.b) of the vehicle for the whole frames of Video 2.

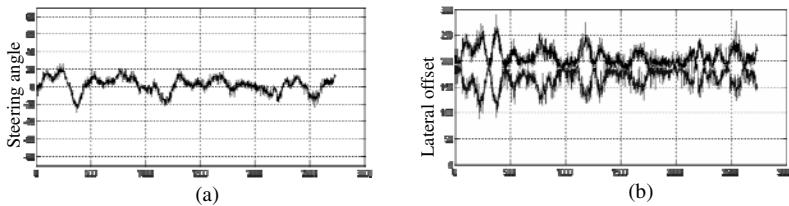


Fig. 5. Evolution of the steering angle (a) and the lateral offset (b)

As illustrated in Figure (5.a), positive values of the steering angle indicate that the vehicle is driving with a deviation of its axis to the right side of the road relative to the middle axis of the lane. On the other hand, negative values indicate a deviation to the left side. The increase of this parameter indicates that the vehicle starts to leave its lane, in which case a warning must be emitted to inform the driver that he is in a prone-situation. The lateral position control is performed also based on the evolution of the lateral offsets of the vehicle, compared to the middle of the lane, as illustrated in Figure (5.b). In this figure, the upper curve corresponds to a remoteness of the vehicle to the right limit and the lower one corresponds to remoteness to the left limit.

Based on these two indicators, a warning must be emitted shortly before the vehicle crosses the line limits. As illustrated in Figure 6, the vehicle began to move out of the lane from the right by the frame number 66 of *Video2*, thus, a warning must be emitted to alert the driver.



Fig. 6. Lane departure warning (Video 2)

5 Conclusion

This paper has a two-fold objective. The first objective is to present an overview of existing approaches to deal with the typical lane detection and tracking steps. Its second objective is to propose a new method and compare its performance with respect to other approaches. Our experimental evaluation showed that, with our method, we can detect lane limits in the majority of the images. With the tracking step, we were able to track the limits during all the frames and to control the lateral position of the vehicle. The findings of the present study are promising. For this reason, studies are currently underway in our laboratories to investigate the detection of obstacles within the tracked lane.

References

1. Dai, X., Kummert, A., Park, S.B., Neisius, D.: A warning algorithm for lane departure warning system. In: IEEE Intelligent Vehicles Symposium (2009)
2. Yu, B., Zhang, W., Cai, Y.: A Lane Departure Warning System based on Machine Vision. In: IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application (2008)
3. Assidiq, A., Khalifa, O., Islam, R., Khan, S.: Real Time Lane Detection for Autonomous Vehicles. In: Int. Conf. on Computer and Communication Engineering, Malaysia (2008)
4. Wang, J., Chen, Y., Xie, J., Lin, H.: Model-based Lane Detection and Lane Following for Intelligent Vehicles. IEEE Intelligent Human-Machine Systems and Cybernetics (2010)
5. Ma, C., Xie, M., Cheng, D.: A Method For Lane Detection Based on Color Clustering. In: 3rd IEEE International Conference on Knowledge Discovery and Data Mining (2010)
6. Zhu, W., Chen, Q., Wang, H.: Lane Detection in Some Complex Conditions. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, China (2006)
7. Wang, Y., Shen, D., Teoh, E.K.: Lane Detection Using Spline Model. Pattern Recognition Letter 21, 677–689 (2000)
8. Tian, M., Fuqiang, L., Wenhong, Z., Chao, X.: Vision Based Lane detection for Active Security in Intelligent Vehicle. In: IEEE Conf. on Vehicular Electronics and Safety, pp. 507–511 (2006)
9. D’Cruz, C., Zou, J.J.: Lane detection for driver assistance and intelligent vehicle applications. In: International Symposium on Communications and Information Technologies (2007)
10. Cheng, H., Jeng, B., Tseng, P., Fan, K.: Lane detection with moving vehicle in the traffic scenes. IEEE Trans. on Intelligent Transportation Systems 7, 571–582 (2006)
11. Gonzalez, J.P., Omit, O.: Lane Detection Using Histogram-Based Segmentation and Decision Trees. In: IEEE Int. Transp. Systems, Conference Proceedings Dearborn (MI), USA (2000)
12. Zu, K.: Realtime Lane Tracking of Curved Local Road. In: Proceedings of the IEEE Intelligent Transportation Systems Conference Toronto, Canada (2006)
13. Tsai, M., Hsu-Yung, C., Yu, C., Tseng, C., Fan, K., Hwang, J., Jeng, B.: Lane Detection Using Directional Random Walks. Int. Conf. on Acoustics, Speech, and Signal Processing (2008)

14. Borkar, A., Hayes, M., Smith, M.T.: Robust Lane Detection and Tracking with Ransac and Kalman filter. In: IEEE International Conference on Image Processing, Egypt (2009)
15. Tian, M., Liu, F., Hu, Z.: Single Camera 3D Lane Detection and Tracking Based on EKF for Urban Intelligent Vehicle. In: IEEE Int. Conf. on Vehicular Electronics and Safety (2006)
16. Asif, M., Arshad, M.R., Yousuf, M., Zia, I., Yahya, A.: An Implementation of Active Contour and Kalman Filter for Road Tracking. International Journal of Applied Mathematics (2006)
17. Wang, Y., Teoha, E.K., Shenb, D.: Lane detection and tracking using B-Snake. Image and Vision Computing 22, 269–280 (2004)
18. Boumediene, M., Ouamri, A., Dahnoun, N.: Lane Boundary Detection and Tracking using NNF and HMM Approaches. In: IEEE Intelligent Vehicles Symposium, Turkey (2007)
19. Ben Romdhane, N., Hammami, M., Ben-Abdallah, H.: An Artificial Vision-based Lane Detection Algorithm. In: Computer Science and its Applications, Jeju, Korea (2009)
20. Ben Romdhane, N., Hammami, M., Ben-Abdallah, H.: A Comparative Study of Vision-based Lane Detection Methods. In: Advanced Concepts for Intelligent Vision Systems, Belgium (2011)

Effective TV Advertising Block Division into Single Commercials Method

Pawel Biernacki

Telecom, Acoustic and Computer Science Institute,
Wroclaw University of Technology,
Wyb. Wyspianskiego 27, 50-350 Wroclaw, Poland
 {pawel.biernacki}@pwr.wroc.pl

Abstract. In this article effective method of TV advertising block division into single commercials using only audio signal is presented. Multidimensional orthogonal signal parametrization scheme was used to set correlation connections in the analyzed signal. Choosing initial potential start/end points of a commercial a computable power was minimized for on-line proposed solution using.

Keywords: commercial separation, orthogonal filter, multidimensional signal representation.

1 Introduction

Jump of the number of the transmitted advertising blocks, in television also, enforce media market researching companies for effective finding of the instruments on forceful separation of a single commercial from the block and its identification. This article deals with the problem of the television advertising break distribution on the single spots. This question is treated in literature as a audio scene separation, understood as a package of sounds about similar characteristic [1], recorded from transmission audio path. For classification of a given signal the following parameters or the group of the parameters are used: amplitude, energy, subband energy, zero crossing rate, cepstrum, melcepstrum coefficients, etc. [2][3][4]. These features are used for signal classification to one of the following classes: speech, music, silence, noise [5] or to distinguish between five different broadcast genres for which semantic borders can be determined [6]. Mentioned methods act with different efficiency and they require computational heavy stock.

Proposed solution uses multidimensional orthogonal audio signal parametrization for features extraction. To minimize computational consumption preprocessing stage is made which finds the start/end single commercial potential points.

2 Proposed Solution

Television advertising block, which consists of the succeeding commercials, can be characterized by the following properties:

- a) there is a short break (about 30ms) after each commercial,
- b) duration of the commercial is a multiple of 5 s (99% cases),
- c) there is one commercial teller (99% cases),
- d) musical cushion is the same melody (99% cases).

Using mentioned properties the algorithm for single commercial separation consist of the following steps:

- step 1: preprocessing - preliminary analysis of signal audio, which detects the silence fragments of the signal (small energy value) which can be the potential start/end point of the commercial
- step 2: signal parametrization around silence fragments
- step 3: around silence periods signal parameters analysis for establishing start/end point of the commercial
- step 4: final acceptance of chosen time points through analysis of distance among chosen in step 3 points (multiplicity 5 second).

2.1 Silence Points Detection

To detect the potential start/end points of a single commercial low short-time energy ratio- LSTER is used, which is defined as:

$$LSTER = \frac{1}{N} \sum_{n=0}^{N-1} (\text{beta} * sE - E(n)) \quad (1)$$

where *beta* (parameter) is a threshold value and

$$sE = \frac{1}{N} \sum_{n=0}^{N-1} E(n) \quad (2)$$

is a mean energy value for the N 25 ms long frames ,and

$$E(n) = \sum_{k=k_0+n*K}^{k_0+(n+1)*K-1} x(k) * x(k) \quad (3)$$

where k_0 is a starting point of the analyzed signal fragment, K is a number of $x(t)$ samples in 25 ms period.

LSTER coefficient means how many of the N frames have the energy below the mean energy value multiply by *beta* parameter. If LSTER is high sufficiently one can say that silence period has taken a stand in analyzed fragment of the signal.

Of course not every detected silence point means start/end point of the commercial. So the steps 3 and 4 are made to avoid faulty detections.

2.2 Signal Parametrization and Start/End Point of a Commercial Selection

Multidimensional orthogonal signal parametrization around silence periods is used for parameters extraction.

Given a vector $|x>_T$ of samples $\{x_0, \dots, x_T\}$ of a time-series (a voice+music signal), observed on a finite time interval $\{0, \dots, T\}$, the estimate of x signal

$$|\hat{x}_N^{\{M\}}>_T \triangleq \mathbf{P}(S_T)|x>_T \quad (4)$$

is the orthogonal projection of the element $|x>_T$ on the space S_T spanned by the following set of the linear and nonlinear observations

$$|X>_T = [|^1X>_T \ |^2X>_T \ \dots \ |^M X>_T] \quad (5)$$

where

$$\begin{aligned} |^m X>_T = & [|x_{i_1} \dots x_{i_m}>_T; i_1 = 0, \dots, N, \\ & i_2 = i_1, \dots, N, \dots, i_m = i_{m-1}, \dots, N] \end{aligned} \quad (6)$$

for $m = 1, \dots, M$. The orthogonal projection operator on $|X>_T$ is defined as

$$\mathbf{P}(S_T) \triangleq |X>_T \langle X|X>_T^{-1} \langle X|T \quad (7)$$

If an ON (generalized; i.e., multidimensional) basis of the space S_T is known, the projection operator on S_T can be decomposed as

$$\mathbf{P}(S_T) = \sum_{j_1=0}^N \mathbf{P}(|r_0^{j_1}>_T) + \dots + \sum_{j_1=0}^N \dots \sum_{j_M=j_{M-1}}^N \mathbf{P}(|r_0^{j_1, \dots, j_M}>_T) \quad (8)$$

where $\mathbf{P}(|r_0^{j_1, \dots, j_m}>_T)$ stands for the orthogonal projection operator on the one-dimensional subspace spanned by the element $r_0^{j_1, \dots, j_m}$, $m = 1, \dots, M$ of an ON basis of the space S_T . Since

$$\mathbf{P}(|r_0^{j_1, \dots, j_w}>_T) = |r_0^{j_1, \dots, j_w}>_T \langle r_0^{j_1, \dots, j_w}|_T \quad (9)$$

the orthogonal expansion of the estimate of the desired signal can be written as

$$\begin{aligned} |\hat{x}_N^{\{M\}}>_T = & \mathbf{P}(S_T)|x>_T = \sum_{j_1=0}^N |r_0^{j_1}>_T \langle r_0^{j_1}|x>_T + \\ & + \dots + \sum_{j_1=0}^N \dots \sum_{j_M=j_{M-1}}^N |r_0^{j_1, \dots, j_M}>_T \langle r_0^{j_1, \dots, j_M}|x>_T \end{aligned} \quad (10)$$

The estimation error associated with the element $|\hat{x}_N^{\{M\}}>_T$ is then

$$|x\varepsilon_N^{\{M\}}>_T \triangleq \mathbf{P}(S_T^\perp)|x>_T = |x>_T - |\hat{x}_N^{\{M\}}>_T \perp S_T \quad (11)$$

The multidimensional signal parametrization problem can be solved by the derivation of a (generalized) ON basis of the estimation space S_T (i.e. calculation of the orthogonal representation (the generalized Fourier coefficients) of the vector $|x\rangle_T$ in the orthogonal expansion (10)).

To derive the desired ON basis of the estimation space S_T , we employ (consult) the following

Theorem 1. *The partial orthogonalization step results from the recurrence relations*

$$|e_{i_1, \dots, i_q}^{j_1, \dots, j_w}\rangle_T = [|e_{i_1, \dots, i_q}^{j_1, \dots, j_w-1}\rangle_T + |r_{i_1, \dots, i_q+1}^{j_1, \dots, j_w}\rangle_T^x \rho_{i_1, \dots, i_q; T}^{j_1, \dots, j_w}] (1 - (\rho_{i_1, \dots, i_q; T}^{j_1, \dots, j_w})^2)^{-\frac{1}{2}} \quad (12)$$

$$|r_{i_1, \dots, i_q}^{j_1, \dots, j_w}\rangle_T = [|e_{i_1, \dots, i_q}^{j_1, \dots, j_w-1}\rangle_T \rho_{i_1, \dots, i_q; T}^{j_1, \dots, j_w} + |r_{i_1+1, \dots, i_q+1}^{j_1, \dots, j_w}\rangle_T] (1 - (\rho_{i_1, \dots, i_q; T}^{j_1, \dots, j_w})^2)^{-\frac{1}{2}} \quad (13)$$

where

$$\rho_{i_1, \dots, i_q; T}^{j_1, \dots, j_w} = - \langle e_{i_1, \dots, i_q}^{j_1, \dots, j_w-1} | r_{i_1, \dots, i_q+1}^{j_1, \dots, j_w} \rangle_T \quad (14)$$

Proof can be found in [9].

The above relations make it possible to construct an orthogonal parametrization (decorrelation) filter, operating directly on the signal samples. The coefficients

$$\rho_{0; T}^{j_1, \dots, j_w} \quad (15)$$

are the generalized (multi-dimensional) Fourier [9] (i.e. Schur-type) coefficients and r_0 are the orthonormal basis of the $x(t)$ signal elements.

Schur parameters ${}^x\rho$ can be interpreted as a correlation coefficients of the parametrized $x(t)$ signal frames around the silence point [9]. Low value of the ${}^x\rho$ for a point k means the frames correlation absence, which can be interpreted like a acoustic scene changing or like a start/end point of the commercial (properties c) and d) of the advertising block allow for such interpretation).

Analyzing the histogram values of the ${}^x\rho$ coefficients for next moments of times (step 3 of the algorithm) start/end points of the commercial can be established (many small values of the ${}^x\rho$ determine that point).

Established start/end points are analyzed according to the distances between them, which should be the multiple of 5 s. (property b) of the advertising block). After this step (step 4) the algorithm finishes its operations.

3 Simulation

Presented solution was implemented in C language for PC computer. Digital satellite TV platforms (POLSAT, CANAL +, PREMIERE) were used for advertising blocks (63 blocks consist of 672 spots) collecting. Signal parameters: sampling frequency 11025 Hz, 8 bits/sample and $\beta = 0.3$ were used.

Two error measures were used for proposed algorithm efficiency: 1) number of wrong start/end points identifications ($M1$), 2) number of omitted start/end points ($M2$)

Table 1. Influence of the filter parameters on the identification effectiveness

Filter parameters	$M1 [\%]$	$M2 [\%]$
$N=10, M=1$	4.4	4.3
$N=10, M=2$	4.1	4.0
$N=40, M=3$	2.3	2.7
$N=50, M=5$	3.3	3.2

N - filter order, M - degree of nonlinearity

Table (I) shows influence of the filter parameters (filter order N and nonlinearity degree M) on identification effectiveness. Increasing values of the filter parameters above some values does not correct the recognition effectiveness, but can make it worse.

For $N = 40$ and $M = 3$ values of parameters the measure $M1=2.3\%$ and $M2=2.7\%$. Figures below show audio signal example, LSTER coefficient in time and the histogram of the $x\rho$ parameters in time.

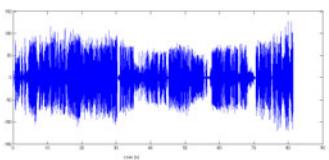


Fig. 1. Audio signal sample

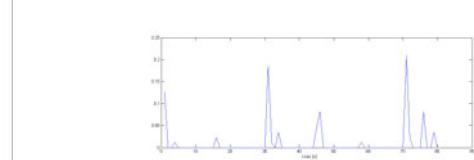


Fig. 2. LSTER value in time for signal

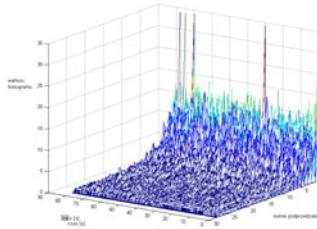


Fig. 3. The histogram of the $x\rho$ in time

Initial signal (figure 1) analysis allowed to detect the silence points (values greater than zero in figure 2). Realized signal parametrization, the $x\rho$ coefficients calculated and their histogram in time established (figure 3) allowed to choose the start/end points of the single commercials (distinctly visible pins on the figure 3). Pins around 80 s detect TV station jingle for advertising block.

In order to compare the proposed solution with existing audio signal identification methods one selected to test the following audio signal parametrization algorithms: Normalized Spectral Subband Moments (NSSM) [8], Mel-Frequency Cepstrum Coefficients (MFCC) [7]. These methods are based on a spectral signal analysis. Selecting $N = 40$, $M = 3$ for presented algorithm the results for

Table 2. Comparison of selected methods

Method	$M1 [\%]$	$M2 [\%]$
NSSM	3.8	6.27
MFCC	4.5	7.36
Proposed method	2.3	2.7

the measures $M1$ and $M2$ are shown in table (2). The best identification was obtained for the proposed solution. For poor quality signals ($fs = 11.025$ kHz 8-bits/sample) algorithms NSSM and MFCC have fared much worse.

4 Conclusion

The presented results allow for the following conclusions:

- efficiency of the algorithm is about 95%.
- proper parametrization filter parameters selection is necessary for the high separation effectiveness
- the algorithm can be used on-line for the TV broadcasting due to the low computational power consumption

References

1. Sundaram, H., Chang, S.: Audio scene segmentation using multiple features, models and time scales. In: IEEE ICASSP 2000. IEEE Computer Society, Washington, DC, USA (2000)
2. Saunders, J.: Real Time Discrimination of Broadcast Speech/Music. In: Proc. ICASSP 1996, Atlanata, GA (1996)
3. Scheirer, E., Slaney, M.: Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator. In: ICASSP 1997, Munich, Germany (April 1997)
4. Zhang, T., Jay Kuo, C.C.: Heuristic Approach for Generic Audio Segmentation and Annotation. In: Proc. ACM Multimedia 1999, Orlando, FL, pp. 67–76 (1999)
5. Lu, L., Jiang, H., Zhang, H.-J.: A Robust Audio Classification and Segmentation Method. ACM Multimedia (2001)
6. Pfeiffer, S., Lienhart, R., Effelsberg, W.: Scene Determination based on Video and Audio Features, REIHE INFORMATIK 20/98, Universität Mannheim (1998)
7. Morgan, N., Bourlard, H., Hermansky, H.: Automatic Speech Recognition: An Auditory Perspective. In: Greenberg, S., Ainsworth, W.A. (eds.) Speech Processing in the Auditory System, p. 315. Springer, Heidelberg (2004) ISBN 9780387005904
8. Paliwal, K.K.: Spectral subband centroid features for speech recognition. In: Proc. IEEE ICASSP, pp. 617–620 (1998)
9. Biernacki, P.: Effective ad recognition using Schur-type signal parametrization. Computer Recognition Systems 2 (2007)

Robust Depth Camera Based Eye Localization for Human-Machine Interactions

Li Li, Yanhao Xu and Andreas König

TU Kaiserslautern, Department of Electrical and Computer Engineering,
Institute of Integrated Sensor Systems,
Erwin-Schrödinger-Str., Gebäude 12, 67663 Kaiserslautern, Germany
{lili,koenig}@eit.uni-kl.de

Abstract. This paper presents a novel approach to depth camera based single-/multi-person eye localization for human-machine interactions. Intensity and depth image frames of a single depth camera are used as system input. Foreground objects are segmented respectively from the depth image by using a novel object segmentation technique based on 2-D histogram with Otsu's method. Contour analysis with ellipse fitting is performed to locate the potential face region on the detected object. Finally, an eye localization algorithm based on a predefined eye template and geometric features is applied on the extracted facial sub-images, which is a hybrid solution combining appearance and feature based eye detection methods using SVM classification to gain robustness. Our goal is to realize a low-cost and robust machine vision system which is insensitive to low spatial resolution for eye detection and tracking based applications, e.g., driver drowsiness detection, autostereoscopic display for gaming/home/office use. The experimental results of the current work with ARTTS 3-D TOF database and with our own Kinect image database demonstrate that the average eye localization rate per face is more than 92% despite of illumination change, head pose, facial expression and spectacles. The performance can be further improved with the integration of an effective tracking algorithm.

Keywords: single-/multi-person eye localization, depth camera, human-machine interactions, foreground object segmentation, geometric features, SVM classification.

1 Introduction

Eye localization has been applied in a wide range of application fields, e.g., autostereoscopic display [21], driver drowsiness detection [20], intelligent video surveillance, human-machine interaction, etc. Traditional eye localization techniques rely on a single data source – intensity (gray-scale) or color image, which have strong dependence on the lighting condition and object reflectivity. For general indoor scenarios foreground object segmentation is another issue which has big impact on the performance of eye localization methods. Stereo imaging system is able to bring distance information to the two dimensional image so

as to facilitate the segmentation procedure. However, stereo imaging has several intrinsic limitations such as correspondence problem (stereo matching) which requires complex algorithm and high computational cost to compensate and stereo imaging itself is also sensitive to illumination change. Current eye localization approaches can be classified in model-based [17], feature-based [16], appearance-based [18] approaches and the combination thereof [15]. Most of the methods are applicable only on (near frontal) facial images, which means extra effort, e.g., for head/face detection, is required to obtain facial images. Facial feature based method such as Gabor response analysis is computational intense. Model-based or appearance-based methods usually can be misguided by the variation of lighting condition.

The motivation of the proposed approach to eye localization has the following aspects: (1) we intend to underline the potential of depth camera in eye detection and tracking based applications and set up an emerging framework for 3-D smart cam based human-machine interactions. (2) We want to utilize depth camera, which is robust to ambient lighting variations, to cope with the problems of conventional intensity-/color-based image segmentation and to substantially facilitate the eye localization procedure.

After a brief introduction, an overview of the state-of-the-art depth cameras is addressed in section 2. The proposed approach to eye localization is discussed in section 3, where we firstly review and motivate depth-based foreground object segmentation and demonstrate the results of our method followed by the detection of potential face region. Then a novel eye localization method based on predefined eye template and geometric features is introduced. The proposed method is validated by presenting the experimental results in section 4. Finally, the current work is concluded in section 5.

2 State-of-the-Art Depth Cameras

In the last decade a type of solid state sensor based on Time-of-Flight (TOF) principle [7,8] has gained more and more attention in machine vision applications. A TOF depth camera with active illumination is capable of simultaneously perceiving reflectance and distance information of objects in a scene at real-time video frame rates. The obtained intensity image and depth image are registered pixel by pixel accordingly and do not require extra effort for image matching which is crucial in conventional stereo imaging systems. In addition, Microsoft launched in 2010 a new depth camera – Kinect for its Xbox 360 video game platform. Similar to TOF depth camera, Kinect has dual image output, i.e., depth and color (RGB) images, which are pixel-aligned through a so-called *Registration* procedure. Depth measurement of Kinect is based on active IR Light CodingTM technology of PrimeSense which is claimed to be immune to ambient light [24]. Due to the standard CMOS technology Kinect becomes a popular mass-production consumer electronics device while maintaining low cost. Hence Kinect is of particular interest for accomplishing our goal. The specifications of several state-of-the-art depth cameras are summarized in Table 1.

Table 1. Specifications of State-of-the-Art Depth Cameras

Manufacturer	Model Name	Depth Accuracy	Frame Rate ¹	Operation Range	Illumination Type	Output Resolution
MESA Imaging	SR4000	1cm	50 fps	0.8 – 8m	NIR 850nm	176 × 144
PMD Tech	CamCube3	1cm	80 fps	0.3 – 7m	NIR 870nm	200 × 200
Fotonic	B70 ²	0.3 – 1.5cm	75 fps	0.1 – 7m	NIR 808nm	160 × 120
Panasonic	D-Imager	3 – 14cm	30 fps	1.2 – 9m	NIR 850nm	160 × 120
Optima	DS10k-A	1 – 3cm	50 fps	1 – 10m	NIR 870nm	120 × 90
MESA Imaging	SR3000 ³	1cm	25 fps	0.8 – 8m	NIR 850nm	176 × 144
Microsoft	Kinect ⁴	1cm	30 fps	0.8 – 3.5m	NIR	640 × 480

1. maximum frame rate
2. the depth camera is based on Canesta Jaguar sensor chip
3. images of ARTTS public database are acquired by SR3000 TOF depth camera
4. Kinect (not TOF based) is used to create our own database

3 Approach to Single-/Multi-Person Eye Localization

The proposed algorithm is outlined in Fig. 1. After a denoising procedure with median filter, foreground object segmentation is performed on the input depth image to extract individual objects from the foreground. Contour analysis is applied on the detected object regions to find potential faces in the scene. Facial sub-images are extracted from the input intensity image based on the face region masks obtained from the previous stage. Eye candidates are further located and paired in group in the normalized face region. In addition, the geometric features of the eye pair candidates are extracted based on an extended eye template. A SVM classifier is employed to classify the eye pair candidates. Finally, the eye location in the facial sub-image is given by the verified eye pair. The eye location in the original image is also retrieved.

3.1 Foreground Object Segmentation Based on Depth Image

Intensity-/color-based object segmentation is vulnerable to the variations of ambient light. Depth-based object segmentation is able to overcome the problem by using active illumination, hence it is less sensitive or even immune to the background illumination. Due to the depth information the segmentation method can achieve better discrimination of different objects. Some well-known object segmentation techniques for depth images are depth histogram based adaptive thresholding [10], region growing and clustering, edge-based methods, mean-shift [6] and kmeans algorithms. First of all, the region growing and clustering methods involve large number of iteration thus result in high computation overhead. Edge-based methods often require closed boundaries of objects which are usually disconnected or fragmented. Also edge-based methods may not be able to segment objects with curved surfaces. Adaptive thresholding on depth histogram is an intuitive solution with low computational cost. However, due to the loss of two dimensional spatial information depth-thresholding is not effective in case

that multiple objects have similar distance away from the camera. Mean-shift and kmeans algorithms are also iterative in nature and have relative strong dependence on parameter settings. In addition, [11] proposed a novel method based on optimal virtual viewpoint finding to segment objects from a depth image with complex scene. This algorithm explores the virtual viewpoint and separate objects from each other iteratively. However, an optimal viewpoint is not always guaranteed. Similar to [11] point cloud segmentation with kernel density estimation [9] intends to utilize spatial information to partition the foreground into different object regions. In this algorithm searching a proper projection plane and the kernel parameter for density estimation as well as the probability threshold for clustering are crucial. Considerable time of iteration is again required for density estimation and region clustering.

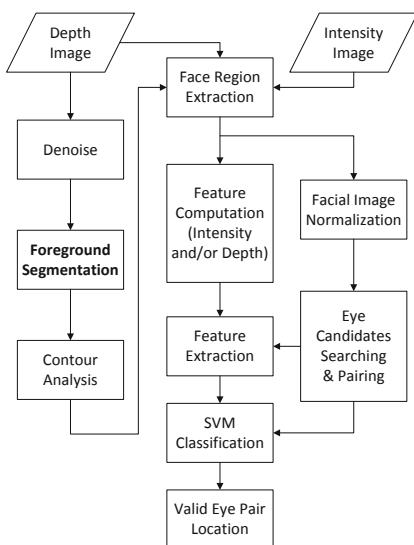


Fig. 1. The processing pipeline of the proposed algorithm

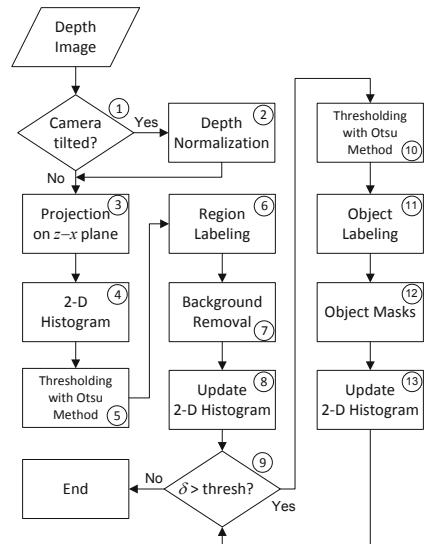


Fig. 2. Flow chart for foreground object segmentation in Fig. 1. δ denotes the maximum value of the updated 2-D histogram, $thresh$ denotes the termination threshold.

In this paper we propose a novel method aimed at foreground object segmentation. At first, if the depth camera is tilted in vertical direction the depth image will be normalized to ensure the foreground objects remain perpendicular to the viewpoint of the camera (see Block 1-2 in Fig. 2). Then the depth image being represented as point cloud is projected on the $z - x$ plane (see Block 3 in Fig. 2), where z indicates the depth value of a single pixel (x, y) in the image.

Large volumetric solid dense objects, e.g., people in the foreground tend to form clusters on the projection plane, while sparse distributed particles,

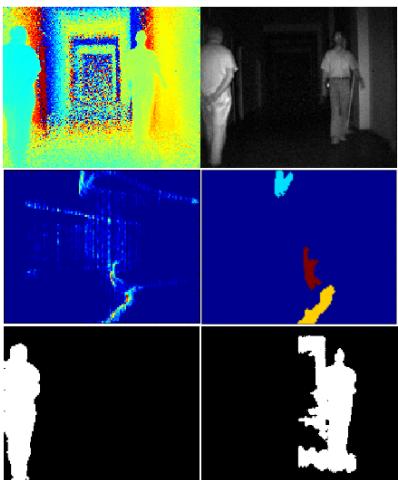


Fig. 3. Foreground object segmentation for ARTTS image left to right and top to bottom, original depth image, original intensity image, point cloud projection on the $z - x$ plane (see Block 3 in Fig. 2), labeled objects (Block 11 in Fig. 2), object masks for two persons in the scene

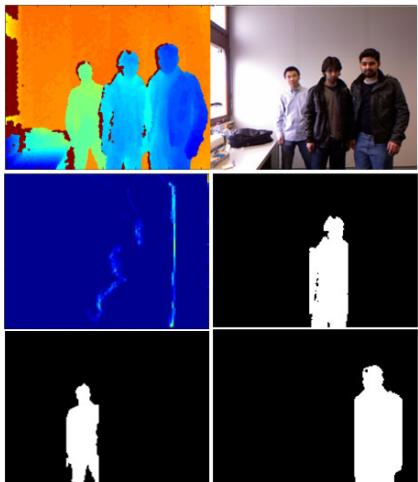


Fig. 4. Foreground object segmentation for Kinect image left to right and top to bottom, original depth image, original RGB image, point cloud projection on the $z - x$ plane (see Block 3 in Fig. 2), object masks for three persons in the scene

e.g., randomly distributed noise or little volumetric objects in the background will remain sparse. Afterwards a 2-D histogram is computed from the projected depth image for further processing (see Block 4 in Fig. 2). This 2-D histogram is able to give globally estimated probability of object positions while preserving the spatial information of object distribution on the projection plane. In our algorithm, Otsu's method is applied on the 2-D histogram of the projected depth image to separate the foreground objects automatically (see Block 5 in Fig. 2). Otsu's method is a nonparametric and unsupervised effective algorithm for intensity (gray-scale) image thresholding [1]. This algorithm assumes that there are two classes of pixels in the image regarding their intensity values and exhaustively searches for the threshold to minimize the intra-class variance. After applying Otsu's method on the 2-D histogram, a global threshold to distinguish densely populated objects and sparse populated particles has been found. This procedure can segment the major objects including the background (if it is densely populated) from the scene. A depth threshold (in z -dimension) is determined by the estimated position of background objects on the projected depth image. The background needs to be removed from the 2-D histogram to avoid misguiding the object segmentation in the foreground region (see Block 6-7 in Fig. 2). The 2-D histogram is then updated by setting the values of segmented regions to zero. The succeeding segmentation procedure is performed in a hierarchical manner that foreground objects are separated by thresholding the

updated 2-D histogram with Otsu's method iteratively (see Block 9-13 in Fig. 2). A certain threshold inferring the object size is set to terminate the iteration which means after segmenting the major parts of the foreground scene the segmentation procedure stops automatically (see Block 9 in Fig. 2). In our experiment, the termination threshold is empirically set to 0.05% of the total pixel count of the depth image. This threshold works properly with both databases. Finally, small size of spurious object regions in the foreground will be discarded. The proposed algorithm for depth-based foreground object segmentation is summarized in Fig. 2. Examples for the segmentation results are demonstrated in Fig. 3 and Fig. 4.

The advantages of the proposed method over state-of-the-art techniques are: (1) the algorithm does not exhaustively search an optimal projection plane which is not always guaranteed. The $z - x$ projection is an efficient solution for most indoor scenarios. (2) The algorithm utilizes thresholding-based method on 2-D histogram to achieve fast segmentation rate. Since multiple objects can be segmented from the scene within a single iteration loop, the overall computational cost is reduced remarkably. In our experiment, the foreground objects in most depth images can be segmented in two iterations despite of slight over-segmentation.

3.2 Face Region Detection and Normalization

Potential face region is detected by applying contour analysis on the object mask generated by the segmentation procedure addressed in section 3.1. Firstly, the concave corners are detected on the object contour. The contour is then split into several fragments between two concave corners. We choose contour fragments from top to bottom of the object mask and apply direct least squares ellipse fitting [12] to locate the face region from the object mask. Constraints such as face size, aspect ratio are applied to verify the potential elliptical region on the mask. Validated elliptical region candidate is marked as face to extract the facial sub-image from the original intensity image.

Generally the extracted facial sub-images have different scales depending on the distance between the user and camera. To apply the predefined template for eye candidates searching the face size normalization is required. Since the original image output of Kinect is with 640×480 spatial resolution, we normalize the facial sub-images in the size of 60 by 50 pixels. The ARTTS images have lower spatial resolution of 176×144 , hence the normalized size of face is determined as 24 by 24 pixels. The located eyes on the normalized facial sub-image through the succeeding processes addressed in section 3.3 and 3.4 can be mirrored back to the original image based on the scale factor as determined here.

3.3 Eye Candidates Searching Based on Predefined Eye Template

In this work a predefined eye template (see Fig. 5) is employed to search the left and right eye candidates in the facial intensity image obtained in section 3.2. The design of the eye template is based on the fact that the iris is darker than the

surrounding sclera in the eye region. Unlike the eye template proposed in [14], the template in our method does not involve the width and height parameters for the eye window due to the low spatial resolution of the face image. In addition, our template highlights the bilateral symmetry of a pair of eyes with respect to the pixel intensity (eye pairs are expected to appear in the scene and occlusion of eyes is not considered here). Thus the facial sub-image is divided in two sub-regions, i.e., the left and right sides, along the major axis (symmetry axis) of the fitted ellipse. The implementation of the eye template is inspired by Monotonie-Operator [3], Harris [4] and SUSAN [5] corner detectors and is defined as follows

$$M_i(x, y) = \begin{cases} 1 & \forall I_j, I(x, y) < I_j \\ 0 & \text{otherwise} \end{cases} \quad i = 1, 2 \quad j = 1, \dots, 5 \quad (1)$$

where $I(x, y)$ corresponds to the intensity value of the current pixel (x, y) , I_j represents the intensity value of its five neighbors. $M_1(x, y)$ and $M_2(x, y)$ are the output maps of eye candidates on the left and right sides of the face, respectively. A final map M for all the eye candidates are generated as

$$M = \bigcup M_i \quad i = 1, 2. \quad (2)$$

To rule out the spurious eye candidates with high intensity values we only consider 50 candidates with the lowest intensity values for the Kinect images and 30 candidates for the ARTTS images, accordingly. In addition, the valid candidates should be located in the upper $\frac{3}{4}$ part of the candidate map M considering possible head pose. Then the following rules are applied to group the eye candidates in pairs: (1) the distance between eye candidates should be greater than $\frac{1}{5}$ face width and less than $\frac{3}{5}$ face width (face width is defined by the length of ellipse minor axis). (2) The distance of eye candidates away from symmetry axis should be within a certain threshold, here $\frac{1}{5}$ face width (symmetry axis is defined by the ellipse major axis). (3) The angle of the connection line of two candidates should be within $[-20, +20]$ degree from the ellipse minor axis. Examples for the paired eye candidates are shown in Fig. 8.

3.4 Computation and Extraction of Geometric Features

The mean (H) and Gaussian (K) curvatures are well-known measures for surface classification. According to the sign of the computed H and K surface curvatures image pixels can be classified as one of the eight different surfaces, i.e., peak, pit, flat, minimal, ridge, valley, saddle ridge and saddle valley. Generally H and K surface curvatures are computed from depth images. Due to the surface coherency property inferred by reflectance information the curvatures may also be estimated from the intensity image [19]. In this work the depth images of the ARTTS and Kinect databases all exhibit poor discrimination among facial components. Hence we prefer to estimate the surface curvatures from intensity images and use the curvature measures as local features for the classification of eye candidates. We simply employ an alternative with e_0 and e_2 , which is proposed in [2], to H and K curvature measures. Here e_0 and e_2 represent the

geometric features (the so-called generalized eccentricities) that correspond to six surface types as depicted in Fig. 7. Since the iris region of an eye tends to be a local minimum regarding the pixel intensity, it is expected to be classified as *pit* in the feature space (see Fig. 7).

After face region detection in section 3.2, a geometric feature map is computed from the intensity values of the facial sub-image using the method implemented in [2]. This procedure can be performed in parallel with eye candidates searching described in section 3.3 to achieve higher processing speed. In this work we select a region based on the original eye template, where 18 pixels are involved including the eye candidates themselves, or based on an extended eye template with 45 pixels in total (see Fig. 6) to extract features for classification purpose. The extended eye template consists of prominent regions corresponding to the potential eyes and nose. The comparison of two feature extraction schemes is addressed in section 4.

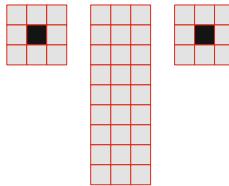
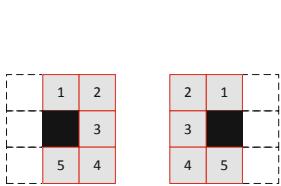


Fig. 5. The predefined eye template for candidates searching

Fig. 6. The extended eye template for feature extraction

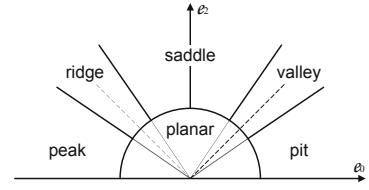


Fig. 7. Six surface types *pit*, *valley*, *saddle*, *ridge*, *peak* and *planar* within the feature space determined by e_0 and e_2 [2]

4 Experimental Results

We use the following databases to evaluate the proposed algorithm: (a) face detection dataset of ARTTS 3-D TOF database [22] containing 260 facial images of 10 different faces with illumination change, (constrained) head pose and facial expression; (b) the Kinect image database created in our institute, which consists of 366 facial sub-images in 171 image frames involving 10 persons with variations in illumination, skin color, head pose, facial expression and spectacles. It should be mentioned that a sensor calibration procedure is required to compensate the misalignment between depth and RGB images of Kinect before performing the proposed algorithm.

The SVM classifier is trained in a supervised manner with hand labeled eye pairs and non-eye-pair candidates (*ground truth*). Depending on the selected template for feature extraction (see Fig. 5 and Fig. 6) and the parameter setting of the SVM classifier the average detection accuracy (ACC) of eye pairs in 30 runs is ranging from 80.44% to 92.08% for ARTTS images. The experimental results are summarized in Table 2.

In the Kinect image database different faces are extracted from images with one up to three persons in the scene by using the method addressed in section

Table 2. Results of Eye Localization with the ARTTS database

Kernel Function	Parameter Setting	Feature Extraction	ACC of 30 runs	False Pos. Rate	False Neg. Rate
Polynomial Function	$order = 3$	eye template	80.44%	28.79%	10.33%
		ext. eye template	87.88%	20.77%	3.46%
Gaussian Function	$\sigma = 3$	eye template	89.88%	11.38%	8.85%
		ext. eye template	92.08%	6.18%	9.67%

3.1 and 3.2. Here, a dataset consisting of 575 true and 575 false eye pairs is selected from the results of eye candidates searching addressed in section 3.3. For holdout validation half of the dataset is randomly picked as training set. The rest of the dataset is regarded as test set. The average detection accuracy for Kinect images in 30 runs has been improved to 94% due to the higher spatial resolution compared to ARTTS images. Multiple pairs of candidates indicating the same eye pair can be detected (see results in Fig. 8) which may be merged within an eye tracking scheme in future work. The experimental results for the Kinect database are summarized in Table 3.

Table 3. Results of Eye Localization with the Kinect database

Kernel Function	Parameter Setting	Feature Extraction	ACC of 30 runs	False Pos. Rate	False Neg. Rate
Polynomial Function	$order = 3$	eye template	89.12%	14.43%	7.33%
		ext. eye template	90.08%	12.94%	6.90%
Gaussian Function	$\sigma = 3$	eye template	86.58%	18.91%	7.93%
		ext. eye template	94.38%	7.92%	3.32%

The processing pipeline of the proposed eye localization method is depicted in Fig. 8 with an example from the Kinect image database. The RGB image is converted to intensity image due to the intensity-based eye candidates searching scheme addressed in section 3.3. Eventually, color-based eye detection techniques can be combined with the current method in future work. Fig. 9 illustrates the eye localization rate per frame for a given Kinect image sequence with two persons in the scene. Due to the imprecise face region detection spurious eye candidates at the neck are misdetected in face 1 of frame No. 13 in the sequence (see Fig. 9 left). Hence the false positive rate (FPR) is high. However, the corresponding false negative rate (FNR) is zero, which means the true eye pair is also detected. For a viable tracking scheme more intelligence is required here to rule out the misdetected eye pair. In face 2 of frame No. 2 the FNR is 100% while the FPR is zero. This problem can be compensated by a predictable tracking algorithm. The worst case with both high FPR and FNR does not occur in our experiment.

To evaluate the intermediate results for face region detection we apply the Viola-Jones face detection method [13] on all 171 Kinect *intensity* images for comparison purpose. Due to the tilted head pose Viola-Jones face detection yields inferior results. In addition, 40 Kinect images with 61 faces in total, which are different from the train/test set mentioned above, are used to compare the

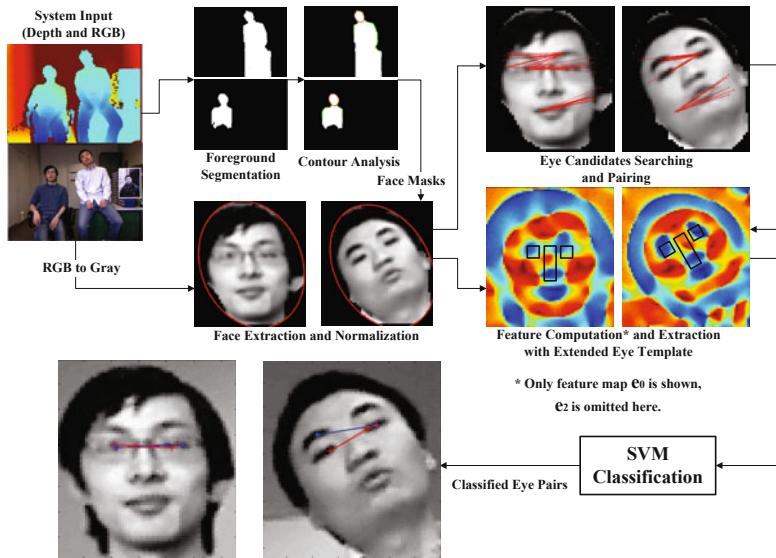


Fig. 8. Experimental results in the processing pipeline with Kinect image. The valid eye pairs are marked with red crosses in the final stage while the solid blue squares in line represent the false positive results.

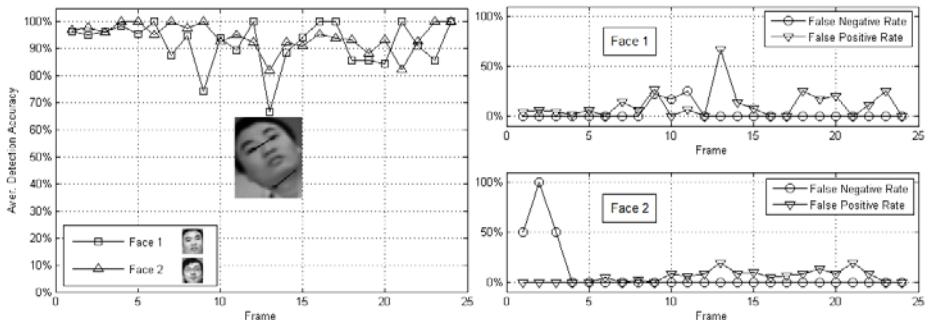


Fig. 9. Eye localization rate per frame for a given Kinect image sequence left to right, average detection accuracy of two faces, FPR/FNR of two faces

eye localization results. Lower tolerance ratio requires higher precision of the detected eye pair position. The proposed algorithm outperforms Ren et al.'s method [14] in all cases due to the insensitivity to low spatial resolution and the SVM classification. The comparison results for face region detection and eye localization are summarized in Table 4.

Table 4. Comparison Results for Face Region Detection and Eye Localization

Face Region Detection	True Pos. Rate	False Pos. Number	Eye Localization	Tolerance = 0.7	Tolerance = 2
Proposed Algorithm	92.62%	72	Proposed Algorithm	84.43%	86.89%
Viola-Jones Algorithm	66.39%	180	Ren et al.'s Algorithm	14.75%	50.82%

5 Conclusion

In this paper, we present an emerging framework for 3-D smart cam based human-machine interactions, e.g., driver drowsiness detection, autostereoscopic display for gaming/home/office use. In this context, we propose an approach to depth camera based single-/multi-person eye localization and future tracking. A novel algorithm for foreground object segmentation is proposed which achieves comparable results to the state-of-the-arts with lower computational cost. The proposed eye template is in particular suitable for images with low spatial resolution. The average detection accuracy of eye pairs on different databases with one up to three persons in the scene is more than 92% despite of the simplified scheme for face detection. False detection of eye pairs can be compensated by an effective tracking algorithm. Potential parameter sensitivity of the proposed algorithm will be addressed in future work. Also, there are several aspects that have potential to further improve the performance of our approach: (1) optimization of the template for feature extraction with regard to geometry properties of the facial components; (2) combination of the geometric features with other local features to achieve better discrimination among the candidates; (3) integration of eye candidates searching with the classifier.

Acknowledgments. The author would like to thank ARTTS consortium for providing the ARTTS 3-D TOF database and to thank OpenKinect community [23] for the *libfreenect* software being used for Kinect image acquisition.

References

1. Otsu, N.: A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. SMC* 9, 62–66 (1979)
2. Haker, M., Böhme, M., Martinetz, T., Barth, E.: Geometric Invariants for Facial Feature Tracking with 3D TOF Cameras. In: ISSCS, pp. 1–4 (2007)
3. Kories, R., Zimmermann, G.: Eine Familie von nichtlinearen Operatoren zur robusten Auswertung von Bildfolgen. In: Ausgewählte Verfahren der Mustererkennung und Bildverarbeitung, pp. 96–119 (1989)
4. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proc. 4th Alvey Vision Conference, pp. 147–151 (1988)
5. Smith, S.M., Brady, J.M.: SUSAN—A New Approach to Low Level Image Processing. *International Journal of Computer Vision* 23(1), 45–78 (1997)

6. Bleiweiss, A., Werman, M.: Fusing TOF Depth and Color for Real-Time Segmentation and Tracking. In: Kolb, A., Koch, R. (eds.) Dyn3D 2009. LNCS, vol. 5742, pp. 58–69. Springer, Heidelberg (2009)
7. Gokturk, S.B., Yalcin, H., Bamji, C.: A Time-Of-Flight Depth Sensor - System Description, Issues and Solutions. In: Conf. on CVPRW, p. 35 (2004)
8. Elkhalili, O.: Entwicklung von optischen 3D CMOS-Bildsensoren auf der Basis der Pulslaufzeitmessung. Dissertation, University of Duisburg-Essen (2005)
9. Akman, O., Bayramoglu, N., Alatan, A.A., Jonker, P.: Utilization of Spatial Information for Point Cloud Segmentation. In: 3DTV Conf., pp. 1–4 (2010)
10. Parvizi, E., Wu, Q.M.J.: Real-Time Approach for Adaptive Object Segmentation in Time-of-Flight Sensors. In: 20th IEEE Int. Conf. on TAI, vol. 1, p. 236 (2008)
11. Merchán, P., Adán, A., Salamanca, S., Cerrada, C.: 3D Complex Scenes Segmentation from a Single Range Image Using Virtual Exploration. In: Garijo, F.J., Riquelme, J.-C., Toro, M. (eds.) IBERAMIA 2002. LNCS (LNAI), vol. 2527, pp. 923–932. Springer, Heidelberg (2002)
12. Fitzgibbon, A.W., Pilu, M., Fisher, R.B.: Direct Least Squares Fitting of Ellipses. In: Proc. of the 13th Int. Conf. on Pattern Recognition, vol. 1, pp. 253–257 (1996)
13. Viola, P., Jones, M.: Rapid Object Detection using a Boosted Cascade of Simple Features. In: Conf. on CVPR, pp. I-511–I-518 (2001)
14. Ren, J.F., Jiang, X.D.: Fast Eye Localization Based on Pixel Differences. In: 16th IEEE Int. Conf. on Image Processing, p. 2733–2736 (2009)
15. Kith, V., El-Sharkawy, M., Bergeson-Dana, T., El-Ramly, S., Elnoubi, S.: A Feature and Appearance Based Method for Eye Detection on Gray Intensity Face Images. In: Int. Conf. on Computer Engineering and Systems, p. 41 (2008)
16. Kim, H., Lee, J.H., Kee, S.C.: A Fast Eye Localization Method for Face Recognition. In: 13th IEEE Int. Workshop on RHIC, pp. 241–245 (2004)
17. Yang, F., Dai, J.W., Liu, D.: A Novel Eye Localization Method Based on Spectral Residual Model. In: 7th World Congress on ICA, p. 6773 (2008)
18. Rurainsky, J., Eisert, P.: Eye Center Localization Using Adaptive Templates. In: Conf. on CVPRW, p. 67 (2004)
19. Besl, P.J., Jain, R.C.: Segmentation Through Variable-Order Surface Fitting. IEEE Trans. PAMI 10, 167 (1988)
20. ElSabrouty, M., Hamdy, A., Fawky, A., Khalil, S.: Drowsy Driver Assistant System. In: IEEE Int. Symposium on SPIT, pp. 176–180 (2009)
21. Yang, J.C., Wu, C.S., Hsiao, C.H., Tsai, R.Y., Hung, Y.P.: Evaluation of an Eye Tracking Technology for 3D Display Applications. In: 3DTV Conf., p. 345 (2008)
22. ARTTS 3D-TOF Database, http://www.artts.eu/publications/3d_tof_db
23. OpenKinect Project, http://openkinect.org/wiki/Main_Page
24. PrimeSense Product Specification, <http://www.primesense.com>

Novel Metrics for Face Recognition Using Local Binary Patterns

Len Bui, Dat Tran, Xu Huang, and Girija Chetty

Faculty of Information Sciences and Engineering,

University of Canberra, ACT 2601, Australia

{Len.Bui, Dat.Tran, Xu.Huang, Girija.Chetty}@canberra.edu.au

Abstract. The paper presents a novel approach to face recognition using Local Binary Patterns (LBP) with the novel soft chi square and soft power metrics. Results of intensive experiments on two public databases, FERET and AT&T, show that these new metrics are efficient and flexible for real-time face recognition applications. They can reduce time performance and also achieve high recognition rate.

Keywords: Face recognition, local binary pattern, chi square metric, Euclidean metric.

1 Introduction

Face recognition is one of hot topics in computer vision that is interested by many researchers as it has a variety of potential applications such as video surveillance. Like other recognition problems, face recognition has two main stages which are feature extraction and recognition. Feature extraction is very important as a face recognition system could achieve high performance on both time and accuracy if suitable features are selected. In early works [1-8], intensity or grey values of pixels are used as features. These features are quite simple so the system is easy to be implemented and often has good time performance. However, these features would be sensible to changes in illumination and pose. As a result, it can reduce the accuracy of the system. To deal with this issue, Gabor features have been included. To some extent, Gabor filter is a simple model of eye, so it is robust to illumination and pose. The works in [8-10] reported that their experiments using Gabor features achieved better results compared to those using traditional features. However, they would be complex and need huge system resources such as memories to store them and take significant computation time. Therefore, it is hard to apply Gabor features to real-time face recognition applications.

In 1994, Ojala *et al.* [11] proposed a novel texture descriptor called Local Binary Patterns (LBP) to analyse the structure of textures. Some years later, they continued to propose the improvement for these features [12, 13]. Since then LBP has become one of the powerful features for texture classification due to its robust and stable properties to illumination and pose. In 2004, Ahonen *et al.* [14] applied LBP into the face recognition problem. In their study, a face image will be divided into small regular

regions and the histogram of LBPs is computed. To measure the similarity between two face images, they used chi square metric. Their experimental results on FERET dataset well outperformed other approaches at that time. Recently some authors [15, 16] have proposed a method to combine LBP and Gabor features. It means that LBP operator will be performed on Gabor images. They reported that they achieved excellent results on FERET dataset. However, it inherits the disadvantages of Gabor features. It costs plenty of time to compute those features. Based on these analyses, we would focus on LBPs of grey images in our study.

There are powerful methods available for face recognition. Principal Analysis Component (PCA) proposed by Turk *et al.* [1, 2] is the well-known subspace method. It would be the de factor baseline and be a simple method to reduce the number of features. There are no strong evidences to support that it would improve the recognition rate. Linear Discriminant Analysis (LDA) proposed by Belhumeur *et al.* [3] is another subspace method. It is used to find the best subspace which can separate classes of individuals. Bayesian method proposed by Moghaddam *et al.* [4] is the first method which applies statistics to face recognition. It was one of the best methods in independent test FERET 1996 [17]. These popular methods measure the similarity score between two face images for recognition purpose. Euclidean and its general Mahalanobis distances are popular scores in these approaches. However, it is really interesting that these metrics do not give good results for *LBP* Histogram features (see experiments in Section 5). A better metric used by Ahonen *et al.* was Chi Square metric [14]. However, it takes plenty of computation time. This drawback would limit the performance of a face recognition system. To some extent, it is a “hard” or “non-linear” metric. Specifically, it is hard to apply metric-based learning methods [18, 19] to the Chi Square metric because most of them are based on the Mahalanobis metric. Therefore, we propose two new metrics to overcome this problem. The first one is soft chi square metric which is the generalised chi square metric and the second one is soft power metric which is used to transform features in Chi Square dominated feature space into new feature space based on Euclidean metric which is appropriate for metric metric-based learning methods. It also has another advantage that can save time performance because Euclidean metric is efficiently implemented in computing software tools such as MATLAB.

The remainder of the paper is organised as follows. The second section will give a brief on LBP and its application in face recognition. The two next sections will present our proposed Soft Chi Square and Power metrics. The fifth section will present details of experimental design and results on two public datasets which are FERET and AT&T. The final section will give brief on our approach and future work.

2 Local Binary Patterns, Histogram Feature and Common Metrics

In this section, we present LBPs and their histogram features. We also mention common metrics which are used for these features.

As mentioned above, *LBP* operator is a powerful tool to encode the micro-structure of texture. *LBP* contains a rich of information on local discriminant features. It is assumed in [14] that each region of individual has a specific structure of texture.

Figure 1 illustrates how to compute the basic *LBP* for the central pixel having grey value of 135. In our study, we used extended *LBP*s called uniform patterns. We use their notation for *LBP* operator, $LBP_{p,R}^{u_2}$ (see [11, 12, 14] for more details).

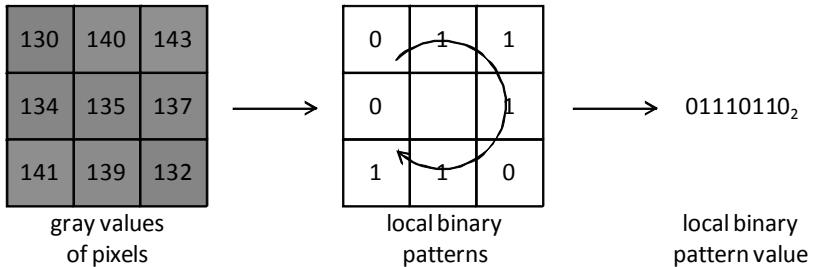


Fig. 1. Illustration of computing *LBP* label of central pixel

However, aligning points is still a big problem in face recognition. Instead of comparing a point with another, most of researchers compare regions. The statistics or histograms of regions are used as features to measure the similarity between two local regions.

$\mathbf{H} = \{H_i, i=1\dots n\}$ is a histogram of a region R of *LBP* image. It can be defined as

$$H_i = \sum_{(x,y) \in R} \{LBP(x,y) = i\} \quad (1)$$

Figure 2 illustrates the framework of feature extraction using *LBP*s. Firstly, *LBP* image of an input image is computed. Next, the area of *LBP* image will be divided into small non-overlapping rectangular regions and histograms of those *LBP* labels are computed using Equation 1. Each bin of histogram characterises the number of specific *LBP* labels. In order to measure the similarity between two histograms of regions, the Min (Eq. 2) and Chi Square (Eq. 3) metrics are used. However, the lack of suitable metrics makes the approach less flexible to combine with other recognition methods.

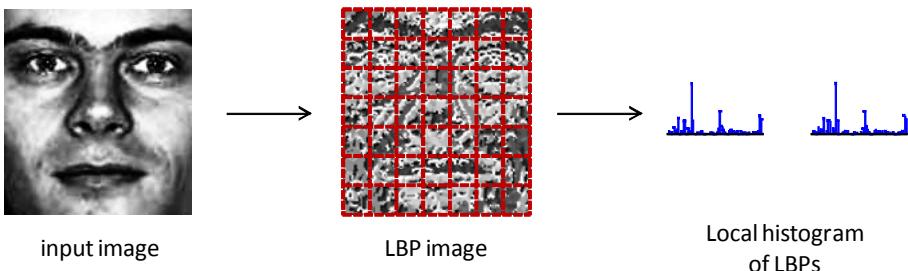


Fig. 2. Illustration of computing histogram of *LBP* labels of an image

$$s_{min}(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^N \min(X_i, Y_i) \quad (2)$$

$$s_{chi}(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^N \frac{(X_i - Y_i)^2}{X_i + Y_i} \quad (3)$$

3 Proposed Soft Chi Square Metric

In this section, we present our proposed soft square metric to solve the limitation of Chi Square distance. As mentioned before, it is noted that there are a lot of metrics available. However, only Min and Chi Square metrics can be successfully applied to face recognition or texture recognition. In fact, the popular Euclidean distance in Eq. 4 usually gives low recognition rates. There is a major difference between Euclidean and Chi Square metrics. The Euclidean-based similarity between two patterns depends only on the difference between features of two patterns. For the Chi Square metric, it depends on not only the difference of features but also their magnitudes. However, there is no mathematical proof to prove that Chi Square is the best metric. Therefore, we propose a novel metric called Soft Chi Square metric defined in Eq. 5. Our metric is a generalise of Chi Square and Euclidean metrics. If $k = 0$, it is Euclidean metric and if $k = 1$, it is Chi Square metric. By using the validation technique on training set D , we can find the suitable k for optimal soft score in Eq. 6.

$$s_{euclidean}(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^N (X_i - Y_i)^2 \quad (4)$$

$$s_{soft\ chi}(\mathbf{X}, \mathbf{Y}, k) = \sum_{i=1}^N \frac{(X_i - Y_i)^2}{(X_i + Y_i)^k}, k \geq 0 \quad (5)$$

$$s_{optimal} = \arg \max_k \left(accuracy(s_{soft\ chi}(\mathbf{X}, \mathbf{Y}, k) | D) \right) \quad (6)$$

4 Proposed Soft Power Metric

Computing the Chi Square similarity score is a hard task. It takes plenty of time for computation. To deal with this problem, we propose a new metric which is called Soft Power. In fact, we transform histogram feature space into new feature space such that Euclidean metric in this space has the same characteristic of Chi Square metric in histogram feature space, and the more magnitude of histogram, the less dissimilarity of patterns. We can choose any function which can keep this characteristic, for example a logarithmic function. In our study, we select the power function defined in Eq. 7 for feature transformation, and the Soft Power score is defined in Eq. 8. Note that our

Soft Power metric is also a generalised Euclidean metric, if $k=1$, it is Euclidean metric. Using the validation technique on training set D, we also find the best k for optimal soft score (Eq. 9)

$$y = f(x) = x^k, k \in [0,1] \quad (7)$$

$$s_{\text{soft power}}(\mathbf{X}, \mathbf{Y}, k) = \sum_{i=1}^N (X_i^k - Y_i^k)^2, k \in [0,1] \quad (8)$$

$$s_{\text{optimal}} = \arg \max_k (\text{accuracy}(s_{\text{soft power}}(k) | D)) \quad (9)$$

5 Experiments

In this section, we present our experiments on two public databases which are AT&T and FERET. We implemented them on MATLAB 2010a and run on our server using Intel Xeon CPU.

5.1 Experiments on AT&T Database

The AT&T database was taken at the University of Cambridge. It contains 400 images of 40 individuals, 37 males and 7 females. One of them has 10 images with a few variations in illumination, pose and expression. We randomly divided the database into two separate subsets for training and testing. The number of images of each set is 200 and each individual has 5 images. We repeated this task four times and determined the average of recognition rates. Overall, we conducted five experiments with different purposes.

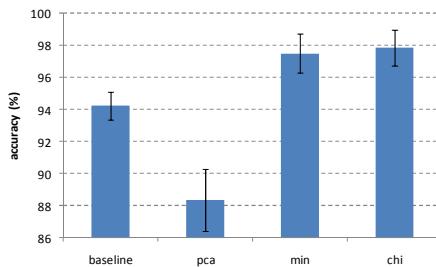


Fig. 3. Recognition results in Experiment 1

Experiment 1: We compared histogram features of LBPs with intensity features in four sub tasks. Task 1 used raw images and Euclidean distance to measure the similarity. Task 2 used PCA (see [1, 2] for more details) to reduce the dimension of images and Mahalanobis distance to compute similarity scores. Tasks 3 and 4 used histogram

of LBPs with operator $LBP_{8,2}^{u^2}$ and window size of 20 by 20, and two metrics Min and Chi Square. Fig. 3 shows the results and it can be seen that the accuracy for using histogram features of LBPs is better than that for using intensity features.

Experiment 2: The size of sub-window has strong effect to the recognition rate. We conducted the experiment to investigate the effect of window size to find out the optimal size. Fig. 4 illustrates recognition results of the experiment. We can see that the optimal size for window is 20 by 20.

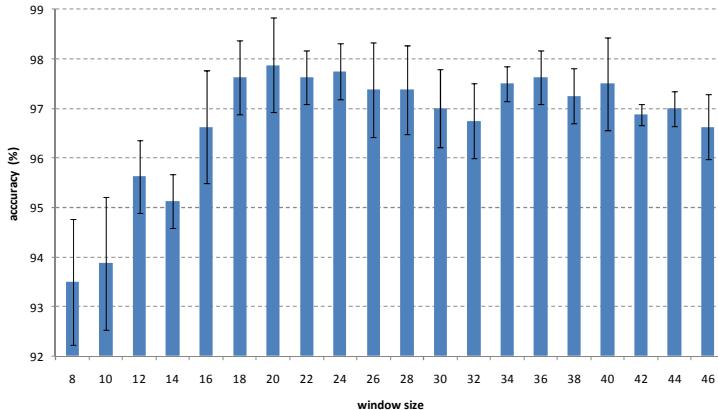


Fig. 4. Recognition results in Experiment 2

Experiment 3: We conducted an intensive experiment to investigate the ability of Soft Chi Square metric. We used 20 by 20 windows and $LBP_{8,2}^{u^2}$ for this experiment and the next two experiments. Fig. 5 shows the results. Note that if $k = 0$, Soft Chi Square distance becomes Euclidean distance; and if $k = 1$, it is Chi Square distance. It proves that we can get the better results with $k = 1.4$.

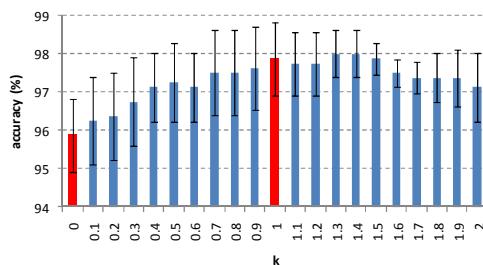


Fig. 5. Recognition results in Experiment 3

Experiment 4: The effect of Soft Power metric is investigated. Fig. 6 illustrates the results. Note that if $k = 1$, it becomes Euclidean metric. According to the results, Soft Power metric can achieve the results of traditional Chi Square metric with $k = 0.5$.

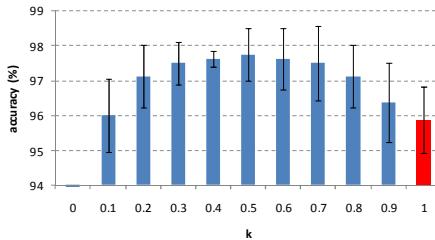


Fig. 6. Recognition results in Experiment 4

Experiment 5: We conducted this experiment to compare the time performance of Min, Chi/Soft Chi Square and Power metrics. It is the average of time to compute the similarity between test set and training set.

Table 1. Time performance (unit: second)

Metric	Average time
Chi/Soft Chi Square	2.187
Min	0.502
Soft Power	0.017

Table 1 contains the results of Experiment 5. The time for Soft Power metric includes the time to do feature transformation using Power function. Obviously, its speed is outperformed other metrics. The reason is that Soft Power metric is Euclidean metric which has efficient implementations on MATLAB.

5.2 Experiments on FERET Database

Grey scale FERET is a standard data set and is widely used for evaluation. There are about 14000 images of more than 1000 individuals. However, we used only five subsets which are Gallery, FB, FC, DUP1 and DUP2.

- Gallery or FA subset contains frontal 1196 images of 1196 people.
- FB subset contains 1195 images. The subjects were asked for an alternative facial expression in FA photograph.
- FC subset contains 194 images. Its images were taken under different lighting conditions.
- DUP1 subset contains 722 images. The photos were taken later in time.
- DUP2 subset contains 234 images. This is a subset of the DUP1 containing those images that were taken at least a year after the corresponding gallery image.

Based on the information containing in ground truth files, we cropped, aligned to normalize images (158 by 186) and apply histogram equalization on them.

Ahonen *et al.* conducted exhaustive experiments on effect of window size and reported the optimal window size and *LBP* operator. We used these parameters in our experiments. We conducted three experiments on FERET dataset.

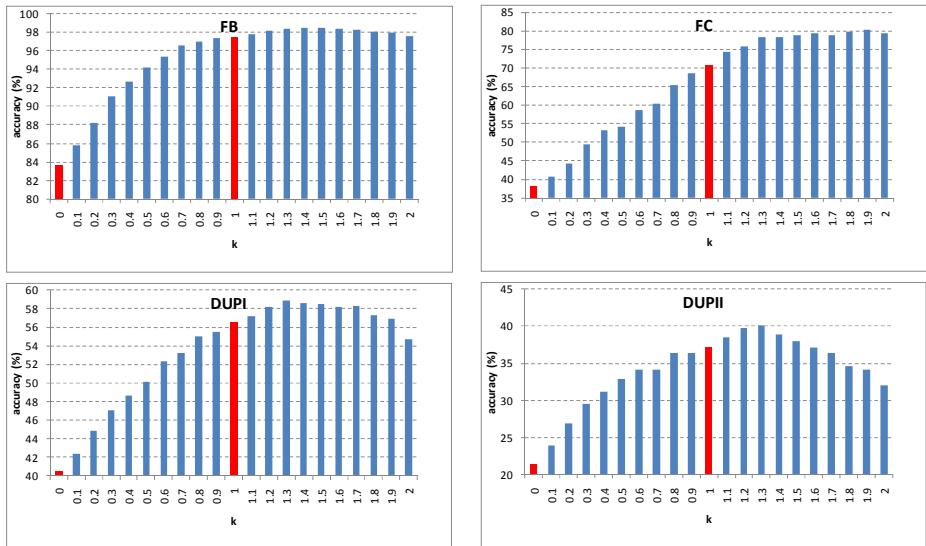


Fig. 7. Recognition results in Experiment 1

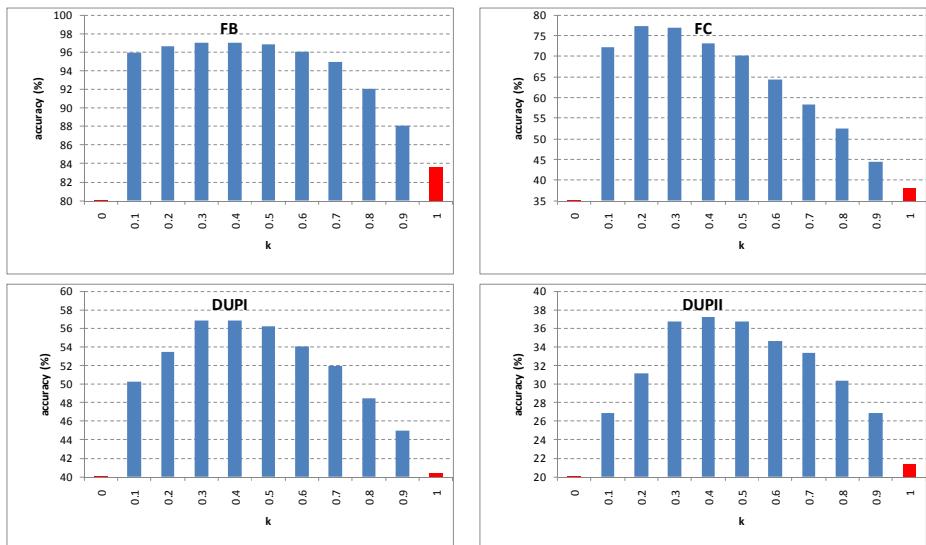


Fig. 8. Recognition results in Experiment 2

Experiment 1: It measured the ability of Soft Chi Square metric. Fig. 7 shows the results of Experiment 1 on four probe subsets. It again proves that Chi Square metric is not an optimal one.

Experiment 2: It investigated the effect of Soft Power metric. Fig. 8 illustrates the results of Experiment 2. It also proves that Soft Power metric gets as good results as Chi Square metric.

Experiment 3: It measured the speed of computation for the following metrics: Min, Chi/Soft Chi Square and Soft Power.

Table 2. Time performance (in seconds)

Metric	FB	FC	DUP1	DUP2
Chi/Soft Chi Square	202.984	32.808	124.903	39.27
Min	52.307	8.044	31.268	9.733
Soft Power	0.52	0.19	0.371	0.192

6 Conclusion

In summary, we have proposed two novel metrics which are Soft Chi Square and Soft Power to deal with problems of time performance and ability of integration due to the lack of metrics. According to our experiments, the results confirmed that both Chi Square and Min metrics are not optimal for LBP histogram features. We have found the optimal metric using Soft Chi Square metric. Especially, Soft Power metric could achieve not only good results but also very high speed on computation. In our future work, Soft Power metric will be combined with metric-based learning methods to get better recognition results.

References

1. Turk, M.A., Pentland, A.P.: Face recognition using eigenfaces. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 1991, pp. 586–591 (1991)
2. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience 3, 71–86 (1991)
3. Belhumeur, P.N., et al.: Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 711–720 (1997)
4. Moghaddam, B., et al.: Bayesian face recognition. Pattern Recognition 33, 1771–1782 (2000)
5. Bartlett, M.S., et al.: Face recognition by independent component analysis. IEEE Transactions on Neural Networks 13, 1450–1464 (2002)
6. Moon, H., Phillips, P.J.: Computational and performance aspects of PCA-based face-recognition algorithms. Perception 30, 303–322 (2001)
7. Phillips, P., et al.: The FERET evaluation methodology for face-recognition algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 1090–1104 (2002)

8. Beveridge, J.R., et al.: The CSU face identification evaluation system. *Machine Vision and Applications* 16, 128–138 (2005)
9. Wiskott, L., et al.: Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19, 775–779 (1997)
10. Peng, Y., et al.: Face recognition using Ada-Boosted Gabor features. In: *Proceedings of Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 356–361 (2004)
11. Ojala, T., et al.: Performance evaluation of texture measures with classification based on Kullback discrimination of distributions, vol. 1, pp. 582–585 (1994)
12. Ojala, T., et al.: A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition* 29, 51–59 (1996)
13. Ojala, T., et al.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 971–987 (2002)
14. Ahonen, T., et al.: Face recognition with local binary patterns. In: Pajdla, T., Matas, J(G.) (eds.) *ECCV 2004. LNCS*, vol. 3021, pp. 469–481. Springer, Heidelberg (2004)
15. Wenchao, Z., et al.: Local Gabor binary pattern histogram sequence (LGBPHS): a novel non-statistical model for face representation and recognition. In: *Tenth IEEE International Conference on Computer Vision, ICCV 2005*, vol. 1, pp. 786–791 (2005)
16. Shan, S., et al.: Ensemble of Piecewise FDA Based on Spatial Histograms of Local (Gabor) Binary Patterns for Face Recognition. In: *18th International Conference on Pattern Recognition, ICPR 2006* (2006)
17. Phillips, P.J., et al.: The FERET evaluation methodology for face-recognition algorithms. In: *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997*, pp. 137–143 (1997)
18. Bar-Hillel, A., et al.: Learning a mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research* 6, 937 (2006)
19. Davis, J.V., et al.: Information-theoretic metric learning, pp. 209–216 (2007)

Unsupervised Scene Classification Based on Context of Features for a Mobile Robot

Hirokazu Madokoro*, Yuya Utsumi, and Kazuhito Sato

Department of Machine Intelligence and Systems Engineering,
Akita Prefectural University,
84-4 Aza Ebinokuchi Tsuchiya, Yurihonjo City, Akita, Japan
madokoro@akita-pu.ac.jp
<http://www.akita-pu.ac.jp/neuro/>

Abstract. This paper presents an unsupervised scene classification method based on the context of features for semantic recognition of indoor scenes used for an autonomous mobile robot. Our method creates Visual Words (VWs) of two types using Scale-Invariant Feature Transform (SIFT) and Gist. Using the combination of VWs, our method creates Bags of VWs (BoVWs) to vote for a two-dimensional histogram as context-based features. Moreover, our method generates labels as a candidate of categories while maintaining stability and plasticity together using the incremental learning function of Adaptive Resonance Theory-2 (ART-2). Our method actualizes unsupervised-learning-based scene classification using generated labels of ART-2 as teaching signals of Counter Propagation Networks (CPNs). The spatial and topological relations among scenes are mapped on the category map of CPNs. The relations of classified scenes that include categories are visualized on the category map. The experiment demonstrates the classification accuracy of semantic categories such as office rooms and corridors using an open dataset as an evaluation platform of position estimation and navigation for an autonomous mobile robot.

1 Introduction

A new lifestyle, including the coexistence of humans and robots in various environments in homes and offices, is anticipated in the near future. For robots to be useful and valuable for the existence of humans, it is necessary that they attain the ability not only to move according to programs installed previously, but also to behave autonomously in many situations and in constantly changing environments. An approach using Simultaneous Localization And Mapping (SLAM) [1] is the mainstream method used to guide automobile movements for a robot to create a map with no human assistance and to estimate its position simultaneously using various sensors to obtain range information: infrared rays, sonar, laser range finders, etc. However, because these sensors can only obtain

* This work is supported in part by a Ministry of Education, Culture, Sports, Science and Technology (MEXT), Grant-in-Aid for Young Scientists (b), No. 21700257.

range data from objects, walls, and obstacles in an environment, it is a challenging task for SLAM-based methods to recognize semantic categories such as kitchens, living rooms, and corridors [2]. We regard the combination of SLAM and semantic scene category recognition as presenting the possibility of creating intelligent and autonomous behavior. Therefore, semantic scene category recognition has attracted attention as an interesting research subject in computer vision and robot vision studies [3].

In computer vision, various methods have been proposed to recognize semantic categories from numerous scene images collected through the internet [4]. However, classification targets are mainly static images of an outdoor environment. Therefore, recognition accuracy drops dramatically for most common indoor scenes when existing methods for outdoor scene classification are tested on indoor scene categories [5]. Human-symbiotic robots are expected to become common in our daily life in the near future. For application of these robots, it is desirable to improve the recognition accuracy against indoor scene categories in our living environments. Robots must have ability based on learning to adapt to an environment that is changed dynamically and momentarily according to human activities and lifestyles. In scene classification and recognition, general and adaptive methods based on machine learning have been proposed according to the progress of computers' calculation performance.

Machine learning is classifiable as supervised learning and unsupervised learning. Training datasets with teaching signals are necessary for supervised learning. The load to collect teaching signals is heavy for robot users. In contrast, unsupervised learning requires no teaching signals during the learning phase. Robots learn the environment by organizing information about the environment: information obtained from various sensors. Users provide semantic information that is assigned to learning results. In contrast to supervised learning, the load for a user is lower when using unsupervised learning. Moreover, robots can discover knowledge from organized information through unsupervised learning. Advanced communication and interaction between robots and humans can be actualized using unsupervised-learning-based methods [6].

This paper presents an unsupervised scene classification method that is based on the context of features. This study is intended to achieve semantic recognition of indoor scenes for an autonomous mobile robot. Our method creates Visual Words (VWs) of two types using Scale-Invariant Feature Transform (SIFT) and Gist. Using the combination of VWs, our method creates Bags of VWs (BoVWs) to vote for a two-dimensional (2D) histogram as context-based features. Moreover, our method generates labels as a candidate of categories while maintaining stability and plasticity together using the incremental learning function of Adaptive Resonance Theory-2 (ART-2). Our method realizes unsupervised-learning-based scene classification using generated labels of ART-2 for teaching signals of Counter Propagation Networks (CPNs). Spatial and topological relations among scenes are mapped on the category map of CPNs. The relations of classified scenes including categories are visualized on the category map. The experiment demonstrates the classification accuracy of semantic categories such as office

rooms and corridors using an open dataset as an evaluation platform of position estimation and navigation for an autonomous mobile robot.

2 Related Work

In context-based scene recognition, features of whole scenes are described after compression in a low-dimensional space based on mechanisms that humans use to recognize scenes. For this approach, the effect of the presence of objects or the precision of segmentation is low because whole-scene information can be described roughly as context. Oliva et al. [7] proposed Gist as a feature to describe global features of a scene. Gist is used popularly in context-based feature description. As scene classification using Gist, Torralba et al. [8] proposed a scene classification method that allocates the number of states on Hidden Markov Models (HMMs) as scene categories. In contrast, Quattoni et al. [5] reported that recognition accuracy drops dramatically for most common indoor scenes when existing methods for outdoor scene classification are tested on indoor scene categories. They specially examined the classification of indoor scenes and proposed a method to improve classification accuracy for indoor scenes. Their method uses a metric function for classifying SIFT features obtained from Regions Of Interest (ROI) and features of Gist on the whole image. However, search results of ROI depend strongly on the classification results because their method requires manual annotation.

3 Context-Based Unsupervised Scene Classification

Figure 1 presents the network architecture used for our method. The procedure consists of the following five steps:

1. feature point detection and description using SIFT,
2. feature description in each block using Gist,
3. creation of 2D histograms,
4. generation of labels using ART-2,
5. and creation of category maps using CPNs.

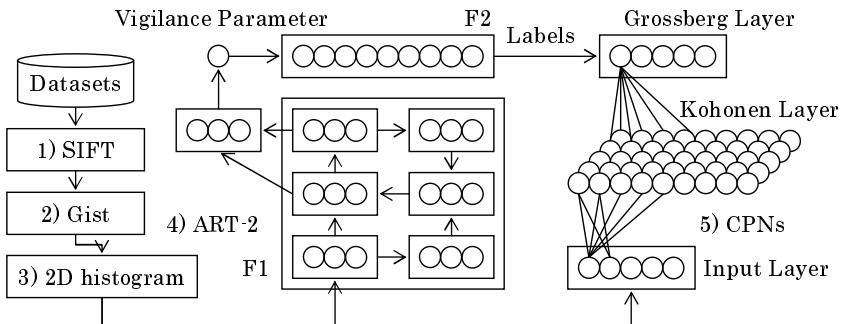


Fig. 1. Overall architecture of our method

Steps 1–3 correspond to creation of BoVWs based on context using SIFT and Gist. Steps 4–5 correspond to unsupervised-learning-based scene classification. Detailed procedures in each step are described as follows.

3.1 Description of SIFT Features

In generic object recognition, SIFT is widely used for describing local features [9]. In object-based scene classification, features of objects in a scene are described using SIFT [4]. Our method uses features obtained using SIFT as foreground features for describing the context.

The SIFT algorithm consist of two steps: feature point extraction and feature description [10]. Actually, Difference of Gaussian (DoG) is used for feature point extraction. A pixel is detected as a candidate of feature points if the attentional pixel that is compared with pixels of 26 neighboring pixels using DoG is selected for the extreme value. Detected feature points as candidates are refined because numerous feature points are included on linear edges. Subsequently, weighted orientation histograms are calculated from the gradient intensity and orientation on surrounding regions of a feature point. In the step of feature description, histograms of eight directions are created in the region of 4×4 blocks. Therefore, 128 dimensional features are calculated. Features of all points are calculated using this procedure.

3.2 Feature Description of Gist

Gist is a general term of semantic scene categories, layout of contained objects, and attribution and knowledge related to primary objects in a scene [11]. The Gist of a scene is characterized as a context that exists for an object in a scene [12]. Our method uses features obtained using Gist as background features for describing context.

Gist is a feature extraction method proposed by Oliva et al. [7]. Primarily, Gist is used for describing structural features in outdoor scenes such as roads, mountains, and buildings. In Gist, frequencies in each block are analyzed using Fourier transformation for dividing regions to $n \times n$ blocks in an image. Moreover, filtering is conducted in each block with cut-off frequencies. Features are extracted to calculate of intensity of arbitrary directional filters for the block after filtering. In our method, we set the number of blocks n is four blocks. The cut-off frequencies are 1, 2, and 4 cycles/image. Orientation filters are 8, 8, and 4 directions in each cut-off frequency. Features are calculated in each color space. Therefore, the feature dimensions per block are 60 dimensions in our method.

3.3 Creation of 2D Histograms

We create 2D histograms as BoVWs. Fig. 2 portrays the procedure for creating 2D histograms. The vertical and horizontal axes respectively portray VWs of Gist and VWs of SIFT. Herein, x_i is an x-coordinate position of a VW on the i -th SIFT feature point. The block of Gist on which the i -th point is located is

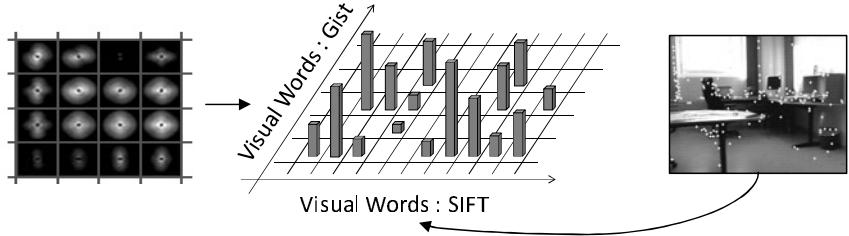


Fig. 2. Generation of a two-dimensional histogram

specialized. Subsequently, y_i is a y-coordinate position of a VW on Gist features. The 2D histogram is created with voting the position of (x_i, y_i) for all SIFT feature points in each image. Our method can describe local and global features as contexts using part-based description of objects as a foreground region and its global feature description as a background region.

Nagahashi et al. [13] proposed a context-based feature description method using 2D histograms. Using their method, using SIFT, foreground features are extracted from annotated object regions and background features are extracted from the region of the scale of six times from each foreground feature. However, their method requires boundaries between foreground and background regions in advance. In contrast, our method can apply images without boundaries between foreground and background regions for mapping SIFT as foreground features and Gist as background features.

3.4 Creation Labels Using ART-2

Actually, ART-2 proposed by Carpenter et al. [15] is a theoretical model of unsupervised neural networks used to form categories for time-series datasets while maintaining stability and plasticity together. Additionally, ART-2 creates labels as a candidate of categories. View images obtained from a mobile robot are changed dynamically according to its movements. We considered that application of ART-2, which can additionally learn time-series datasets is useful for scene classification by a mobile robot.

The network of ART-2 consists of two fields: Field 1 (F1) for feature representation and Field 2 (F2) for category representation. The F1 consist of sub-layers. The sub-layers actualize Short Term Memory (STM), which enhances features of input data and removes noise for a filter. The F2 actualizes Long Term Memory (LTM) based on finer or coarser recognition categories. In this study, we use these categories as labels.

The learning algorithm of ART-2 is the following. Points F1 and F2 are connected via the sub-layer p_i . Input data I_i are presented to F1. After propagating F1, the maximum active unit T_J is searched as

$$T_J(t) = \max\left(\sum_j p_i(t) Z_{ij}(t)\right). \quad (1)$$

Then top-down weights Z_{ji} and bottom-up weights Z_{ij} are updated as shown below.

$$\frac{d}{dt}Z_{ji}(t) = d[p_i(t) - Z_{ji}(t)] \quad (2)$$

$$\frac{d}{dt}Z_{ij}(t) = d[p_i(t) - Z_{ij}(t)] \quad (3)$$

The vigilance threshold ρ is used to judge whether input data correctly belong to a category.

$$r_i(t) = \frac{u_i(t) + cp_i(t)}{e + \|u\| + \|cp\|} \quad \frac{\rho}{e + \|r\|} > 1. \quad (4)$$

The active unit is reset and goes back to the searching step again if eq. 4 is true. Repeat propagation in F1 until the change of F1 is sufficiently small if eq. 4 is not true.

3.5 Category Map Formation Using CPNs

The CPNs proposed by Nilsen [16] are supervised and self-organizing neural networks combine Kohonen's competitive learning [14] algorithm and Grossberg's outstar learning algorithm. The network comprises three layers: an input layer, a Kohonen layer, and a Grossberg layer. Our method uses CPNs for unsupervised learning to provide labels created by ART-2 for teaching signals to the Grossberg layer. The CPNs perform automatic labeling with this mechanism. Our method can create labels as a candidate of a category without setting the number of categories in advance. Moreover, our method can visualize spatial relations among categories based on their similarities.

The CPN learning algorithms are the following. $u_{n,m}^i(t)$ are weights from an input layer unit i ($i = 1, \dots, I$) to a Kohonen layer unit (n, m) ($n = 1, \dots, N, m = 1, \dots, M$) at time t . Therein, $v_{n,m}^j(t)$ are weights from a Grossberg layer unit j to a Kohonen layer unit (n, m) at time t . These weights are initialized randomly. The training data $x_i(t)$ show input layer units i at time t . The Euclidean distance $d_{n,m}$ separating $x_i(t)$ and $u_{n,m}^i(t)$ is calculated as

$$d_{n,m} = \sqrt{\sum_{i=1}^I (x_i(t) - u_{n,m}^i(t))^2}. \quad (5)$$

The unit for which $d_{n,m}$ is the smallest is defined as the winner unit c as

$$c = \operatorname{argmin}(d_{n,m}). \quad (6)$$

Here, $N_c(t)$ is a neighborhood region around winner unit c . In addition, $u_{n,m}^i(t)$ of $N_c(t)$ is updated using Kohonen's learning algorithm, as

$$u_{n,m}^i(t+1) = u_{n,m}^i(t) + \alpha(t)(x_i(t) - u_{n,m}^i(t)). \quad (7)$$

In addition, $v_{n,m}^j(t)$ of $N_c(t)$ is updated using Grossberg's outstar learning algorithm as

$$v_{n,m}^j(t+1) = v_{n,m}^j(t) + \beta(t)(t_j(t) - v_{n,m}^j(t)). \quad (8)$$

Table 1. Settings values of parameters on ART-2 and CPN used in the experiment

		SIFT	Gist	Our Method
ART-2	θ	0.1	0.1	0.1
	ρ	0.80	0.80	0.95
CPN	α	0.5	0.5	0.5
	β	0.5	0.5	0.5
	I	10,000	10,000	10,000

In that equation, $t_j(t)$ is the teaching signal to be supplied to the Grossberg layer. Furthermore, $\alpha(t)$ and $\beta(t)$ are the learning rate coefficients that decrease concomitantly with the learning progress. The learning of CPNs repeats up to the learning iteration that was set previously.

4 Experimental Results Obtained Using KTH-IDOL Datasets

The Image Database for rObot Localization (KTH-IDOL) dataset [17] is an open image dataset used for navigation, localization, and position estimation for a mobile robot in an indoor environment. This dataset is used as a benchmark dataset in indoor scene recognition. Moreover, this dataset is used as a part of Image Cross Language Evaluation Forum (CLEF) 2009 [18]. In this experiment, we evaluated the classification accuracy of semantic scene categories using this dataset.

4.1 Experimental Conditions

The KTH-IDOL dataset comprises time-series images of three weather and illumination conditions: cloudy, night, and sunny. The images were obtained using two robots: Dumbo and Mannie. Mannie is taller than Dumbo. We used images obtained using Mannie under sunny weather conditions. Target scenes are of five categories: Printer Area (PA), One-person Office (EO), Two-person office (BO), Kitchen (KT), and Corridor (CR).

For this experiment, we compared the classification accuracy obtained using three feature representation methods: BoVWs with SIFT, BoVWs with Gist, and BoVW with SIFT and Gist. Table 1 portrays parameters of ART-2 and CPNs used for the three feature representation methods. We set the same values of the initial values of learning coefficients α and β and learning iteration I . We set the common setting value for parameter θ related to the noise rejection of ART-2. Based on preliminary experiments, we set each value of the vigilance parameter ρ that controls the classification granularity.

4.2 Category Formation Results

Figure 3 depicts label formation results with ART-2 for each feature representation. Vertical and horizontal axes respectively show frames of input images

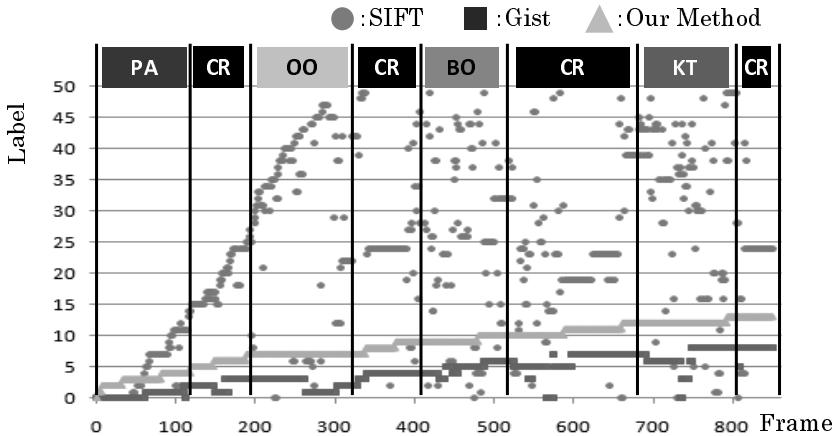


Fig. 3. Results of generated labels using ART-2

and labels of ART-2. The upper part of the graph shows Ground Truth (GT) categories in each scene. The total quantities of labels created by ART-2 are: 50 labels for SIFT, 9 labels for Gist, and 14 labels for Our method. The result for Our method is that labels were generated step-by-step according to semantic categories of scenes. Redundant labels are generated as a result of SIFT. Especially, labels of BO and KT are overlapped. In the Gist result, numerous overlapped labels are generated, although the labels are fewer than those obtained with the result of Our method. Features using Gist alone can not describe indoor scenes because of the global description.

Figure 4 depicts category maps in each feature representation. We set the category map size as 30×30 units. In the SIFT result, labels on the category map are confused because the generated labels are numerous. In the Gist result, scene images that are distributed in several regions produce a confused distribution, although the labels are only nine labels. In contrast, our method formed a category map according to local scenes based on semantic categories. Moreover, no independent labels or confused labels are apparent when using our method.

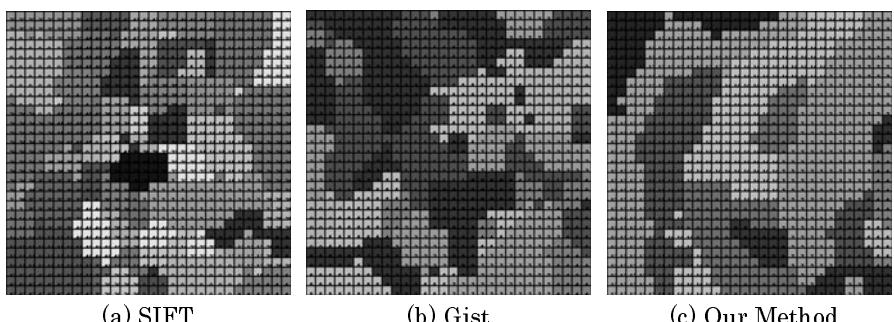


Fig. 4. Comparison of category maps' results obtained using SIFT, Gist, and Our method

Table 2. Comparison of recognition accuracy results.

Method	PA	EO	BO	KT	CR
SIFT	92%	72%	0%	0%	40%
Gist	60%	54%	53%	24%	37%
Ours	96%	98%	67%	86%	78%

4.3 Classification Accuracy

Herein, we use the following recognition accuracy for quantitative evaluation of the classification performance of our method.

$$(RecognitionAccuracy) = \frac{(CorrectData)}{(AllData)} \times 100. \quad (9)$$

Table 2 portrays a comparison of classification accuracy results. The classification accuracy of our method is higher than that of either of the other two methods. The classification accuracy of PA and EO respectively reached 96% and 98%. The classification accuracies of BO and CR respectively remain at 67% and 78%. The mean classification accuracies for each feature representation of SIFT, Gist, and Our method are, respectively, 41%, 46%, and 85%. The result obtained using our method is 44% higher than that of SIFT and 38% higher than that of Gist.

5 Conclusion

This paper presented an unsupervised scene classification method using SIFT and Gist features as a context for semantic recognition of indoor scenes using an autonomous mobile robot. Our method represents spatial relations among categories for mapping neighborhood units on category maps of CPNs while maintaining sequential information using labels generated from ART-2. In the scene classification experiment using the KTH-IDOL dataset, the result of our method using 2D histogram created by SIFT and Gist is superior to the results obtained using SIFT or Gist separately. Our method is effective for indoor scene classification in robot vision.

We will decide a suitable number of categories from extracting category boundaries from the category map on CPNs. Moreover, we must extend the application of our method to a dynamic environment and an environment in which numerous pedestrians work and live.

References

1. Dissanayake, G., Newman, P., Clark, S., Durrant-Whyte, H.F., Csorba, M.: An experimental and theoretical investigation into simultaneous localization and map building (SLAM). In: Experimental Robotics VI. LNCIS. Springer, Heidelberg (2000)

2. Wu, J., Rehg, J.M.: CENTRIST: A Visual Descriptor for Scene Categorization. *IEEE Trans. Pattern Analysis and Machine Intelligence* (2010)
3. Wu, J., Christensen, H.I., Rehg, J.M.: Visual Place Categorization: Problem, Dataset, and Algorithm. In: Proc. IEEE/RSJ Int'l. Conf. Intelligent Robots and Systems (2009)
4. Siagain, C., Itti, L.: Rapid Biologically Inspired Scene Classification Using Features Shared with Visual Attention. *IEEE Trans. Pattern Analysis and Machine Intelligence* 29(2), 300–312 (2007)
5. Quattoni, A., Torralba, A.: Recognizing Indoor Scenes. In: Proc. Computer Vision and Pattern Recognition (2009)
6. Tsukada, M., Utsumi, Y., Madokoro, H., Sato, K.: Unsupervised Feature Selection and Category Classification for a Vision-Based Mobile Robot. *IEICE Trans. Inf. & Sys.* E94-D(1), 127–136 (2011)
7. Oliva, A., Torralba, A.: Building the gist of a scene: the role of global image features in recognition. In: *Visual Perception, Progress in Brain Research*, vol. 155 (2006)
8. Torralba, A., Murphy, K.P., Freeman, W.T., Rubin, M.A.: Context-Based Vision System for Place and Object Recognition. In: Proc. IEEE Int'l. Conf. Computer Vision, pp. 1023–1029 (October 2003)
9. Yanai, K.: The Current State and Future Directions on Generic Object Recognition. *IPSJ SIG Notes CVIM*, 121–134 (September 2006)
10. Lowe, D.G.: Object Recognition from Local Scale-Invariant Features. In: Proc. IEEE International Conference on Computer Vision, pp. 1150–1157 (1999)
11. Torralba, A.: How many pixels make an image? *Visual Neuroscience* 26, 123–131 (2009)
12. Takeuchi, T.: Underlying Mechanisms of Scene Recognition and Visual Search. *ITE Technical Report* 33(24), 7–14 (2009)
13. Nagahashi, T., Ihara, A., Fujiyoshi, H.: Tendency of Image Local Features that are Effective for Discrimination by using Bag-of-Features in Object Category Recognition. *IPSJ SIG Notes DVIM* (3), 13–20 (2009)
14. Kohonen, T.: Self-Organizing Maps. Springer Series in Information Sciences (1995)
15. Carpenter, G.A., Grossberg, S.: ART 2: Stable Self-Organization of Pattern Recognition Codes for Analog Input Patterns. *Applied Optics* 26, 4919–4930 (1987)
16. Hatch-Nielsen, R.: Counterpropagation networks. In: Proc. of IEEE First Int'l. Conf. on Neural Networks (1987)
17. Luo, J., Pronobis, A., Caputo, B., Jensfelt, P.: The KTHIDOL2 database. Technical Report CVAP304, Kungliga Tekniska Hoegskolan, CVAP/CAS (October 2006)
18. Pronobis, A., Xing, L., Caputo, B.: Overview of the CLEF 2009 Robot Vision Track. In: Peters, C., Caputo, B., Gonzalo, J., Jones, G.J.F., Kalpathy-Cramer, J., Müller, H., Tsikrika, T. (eds.) *CLEF 2009. LNCS*, vol. 6242, pp. 110–119. Springer, Heidelberg (2010)

A Novel Emotion Recognizer from Speech Using Both Prosodic and Linguistic Features

Motoyuki Suzuki¹, Seiji Tsuchiya², and Fuji Ren¹

¹ Institute of Technology and Science, The University of Tokushima,
2-1 Minamijosanjima-cho, Tokushima, 770-8506, Japan
moto@m.ieice.org, ren@is.tokushima-u.ac.jp

² Department of Intelligent Information Engineering and Sciences,
Doshisha University,
Kyotanabe, Kyoto, 610-0394, Japan
sttsuchiy@mail.doshisha.ac.jp

Abstract. Emotion recognition based on speech characteristics generally relies on prosodic information. However, utterances with different emotions in speech have similar prosodic features, so it is difficult to recognize emotion by using only prosodic features.

In this paper, we propose a novel approach to emotion recognition that considers both prosodic and linguistic features. First, possible emotions are output by clustering-based emotion recognizer, which only uses prosodic features. Then, subtitles given by the speech recognizer are input for another emotion recognizer based on the “Association Mechanism.” It outputs a possible emotion by using only linguistic information. Lastly, the intersection of the two sets of possible emotions is integrated into the final result.

Experimental results showed that the proposed method achieved higher performance than either prosodic- or linguistic-based emotion recognition. In a comparison with manually labeled data, the F-measure was 32.6%. On the other hand, the average of F-measures of labeled data given by other humans was 42.9%. This means that the proposed method performed at 75.9% in relation to human ability.

Keywords: Emotion recognition, prosodic feature, linguistic feature, association mechanism.

1 Introduction

In recent years, various speech recognition systems have been developed (e.g. [\[III, II\]](#)). These systems can process a diverse vocabulary, grammatical structure and common expressions in order to understand the meaning of utterances. Moreover, advanced robotics technology has brought us closer to the day when we can converse with humanoid robots.

However, these systems only understand the words that the speaker is using. In human communications, the listener can understand what the speaker

is saying plus what he actually means as well as what he is feeling. Speech signals augment the linguistic information with para-linguistic and non-linguistic information [5], which helps the listener to better understand the intention of the speaker. This process is very important for natural conversation. Thus, to realize a more natural conversation between humans and robots, focus is placed on emotion recognition from speech signals.

In many of the emotion recognition systems that have been developed (e.g. [6,14]), recognition is based on extracting the prosodic features from the speech signals, and then estimating the emotion using statistical models. It is well known that prosodic features are strongly related to the speaker's emotions. For example, "hot anger" changes the prosody to a higher pitch, louder voice, and more rapid utterance. "Sadness" changes the prosody to a higher pitch, but a softer voice and slower speaking.

However, different emotions can have the same prosodic features. For example, the prosodic features of "excitement" are very similar to those of "hot anger." "Fear" is similar to "sadness." There are many emotions, but a relatively small number of differences in prosody, which makes it difficult to recognize the speaker's emotions based only on prosodic features.

In the meantime, extensive research is also being conducted on emotion recognition based on linguistic text (e.g. [16,9,22]). There are many key phrases corresponding to emotions in text. For example, "happy," "enjoy," and "win" correspond to the emotion "joy", and "sad," "lose," and "make a mistake" correspond to "sadness." An emotion recognition system finds the key phrases from the input text, and estimates the final emotion by considering the emotions of key phrases, negative words, conjunctions, and other linguistic information.

However, the estimation performance of this type of system is low because the emotion in a sentence can be different from what is depicted by its emotional key phrase. For example, what emotion is indicated by "You are the winner"? If the words are spoken rapidly and at a higher pitch and slightly loud, then the emotion is "joy." However, if the words are spoken softly and slowly and at a lower pitch, because the speaker happens to be the losing opponent, then the emotion is "sadness." Thus, it is difficult to recognize the speaker's emotion based only on linguistic information.

As noted above, the use of either prosodic or linguistic features alone cannot perfectly estimate the speaker's emotions. However, their integration can compensate for each system's shortcomings. Emotions with the same prosodic features can be differentiated using emotional key phrases, and ambiguity of emotions in linguistic information can be resolved by using prosodic features.

Emotion recognition systems designed to use both prosodic and linguistic information can be classified into two types according to the integration method. The first type [21,18] uses "feature level" integration. Two feature vectors are calculated from the prosodic and linguistic information independently, and these vectors are concatenated and input into the classifier. In this type of system, linguistic information must be represented by a vector. This means that only low-level features (e.g. Bag-of-Words, n -gram statistics, key phrase frequency)

can be used. It is difficult to use a “deep” estimation method, such as rule-based deduction or concept-based analysis. Moreover, this type requires a huge amount of training samples for the final classifier because the training data should contain all combinations of prosodic and linguistic variations.

The other type of system [17,19,12] uses “result level” integration. Confidence scores for each emotion are calculated according to prosodic/linguistic emotion recognition independently, and the two scores are combined to select the final result. This integration type is reasonable, but it is important how the scores are combined. Multilayer perceptron has been used for combination [17,19], but this also requires a huge amount of training data as same as the first integration type of system.

The product of the two scores has been used as the final score [12]. However, this score does not consider the characteristics of the linguistic score. As mentioned above (example “You are the winner”), a key phrase is related to a emotion, but it can be used with different emotions. Therefore, linguistic-based emotion recognition outputs a very low score for other emotions, even though prosodic-based emotion recognition outputs a high score for several emotions. As a result, it is not reasonable to use the average score of the two outputs for the final score.

In this paper, we propose a novel approach to emotion recognition based on speech signals. Both prosodic and linguistic features are extracted, and possible emotions are estimated from the two types of features independently. Finally, the two estimated results are combined into the final result by considering the characteristics of linguistic-based emotion recognition. The proposed method decreases the ambiguity of emotions by combining both prosodic and linguistic information.

2 Emotion Recognition Based on Prosodic and Linguistic Features

The proposed emotion recognition process has four parts (Fig. 1). First, the speaker’s utterance is input to the prosodic-based emotion recognition agent (EM_p). The prosodic features are calculated and the possible emotions are output. At the same time, the utterance is also input to the speech recognizer and the results are transferred to the linguistic-based emotion recognition agent

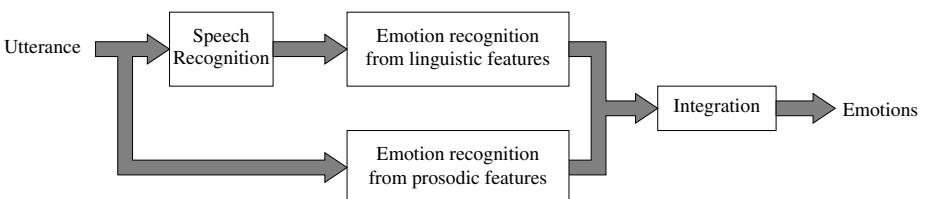


Fig. 1. Block diagram of the proposed method

(EM_l). Finally, the integrator outputs the final emotion by combining the results obtained by the EM_p and EM_l .

Speech recognizers, such as the one employed in this system, are not perfect. Accuracy given by a recent speech recognizer [11] was about 80% for spontaneous speech [15]. To eliminate the risk of misrecognized words adversely affecting the performance of EM_l , subtitles for input speech were used instead of the recognition results in the experiments described in Section 3.

2.1 Emotion Recognition Based on Prosodic Features

There are many emotion recognition systems based on prosodic features [6,14], but the methods used in these systems are similar. First, several prosodic features (pitch, power, etc.) are extracted from the input speech, and numerous statistical parameters (average, standard deviation, maximum, etc.) are calculated from the extracted features. Then, a statistical classifier (neural networks, support vector machines, Gaussian mixture models, etc.) is used for the classification of emotions.

In the proposed system, it is not necessary to determine the final emotion from the prosodic features. As mentioned in Section 1, speech signals for different emotions may have similar prosodic features, therefore it is not relevant to use a statistical classifier (NN, SVM, GMM, etc.) because these classifiers try to separate speech signals with different emotions even if they have similar prosodic features.

In this study, clustering-based method is used for emotion recognition. First, all training samples are clustered using the LBG [13] algorithm. This step is the same as vector quantization (VQ), and the codebook size (number of clusters) is given by humans in advance. After that, an emotion set is defined for each cluster by taking the union of emotions that corresponds to the training samples included in the cluster. In the recognition phase, the distance between the test sample and the centroid of a cluster is calculated for all clusters, and the emotion set that corresponds to the nearest cluster is output as the recognition result.

In this method, supervised signals (“correct” emotion for each training sample) are not needed in the clustering step. This means that clustering is carried out based only on the similarity of prosodic features. In other words, two training samples that have similar prosodic features are clustered into the same cluster even if the samples have different emotions. If the cluster is the nearest one to the test sample, then both emotions are output as a recognition result.

2.2 Emotion Recognition Based on Linguistic Features

Among the many emotion recognition systems based on linguistic information [16,9,22], the majority is based on key phrases. These systems use a dictionary that contains key phrases paired with possible emotions. The performance of this type of emotion recognition system depends on the quality of the dictionary. However, it is difficult to construct a substantial dictionary, which means that the topics and words of the input speech must be limited.

To solve this problem, the proposed system uses the emotion recognition system based on the “Association Mechanism.” [25][23][24] This method also uses a dictionary; however, it calculates the degree of association [27] between an input word and the words registered in the dictionary by using the “Concept Base” [7][10] in order to deal with “unknown words.” In this way, an unknown word can be translated into a known word by using the “Concept Base”, and then the degree of association can be calculated.

Briefly, the method is as follows: first, five components (“subjects,” “modifiers,” “objects,” “action words,” and “linguistic modality”) are extracted from the input text. “Object” words are classified into 203 “sense” by the sense judgment system [8][26], and “action words” are classified as either “succession” or “opposite.” Emotion recognition rules are defined for all combinations of 203 sense words and two types of action words. After emotion recognition, the emotion may be changed by considering “subjects” and “linguistic modality” types. “Subjects” are classified into 27 categories, and “linguistic modality” into 20 categories. Emotion changing rules are defined for all combinations of categories and emotions. In total, the system has 735 recognition rules.

2.3 Integration Method of Two Results

After recognition by the EM_p and EM_l , the two results are integrated into the final result. The intersection of the two results is used as the integration method because the two recognition agents have output all possible emotions. In fact, the EM_l always outputs only one emotion, which means that the intersection of the two results includes no more than one emotion. If the intersection is empty, then the result from EM_p is output as the final result.

3 Experiments

To investigate the effectiveness of the proposed method, emotion recognition experiments were carried out.

3.1 Training and Testing Samples

To investigate the performance using natural conversation, we used the speech signals from a Japanese movie. Several dialog scenes were selected, and 315 speech samples uttered by 15 speakers (11 male and 3 female adults, 1 female child) were extracted. Details are shown in Table II.

For all samples, the “correct” emotion was labeled manually by 44 evaluators. First, the movie was played from the beginning, and then was paused at the selected dialog scene. Evaluators labeled the speaker’s emotion for each selected utterance, and then the movie was continued until reaching the next selected scene.

In this experiment, 7 basic emotions (“joy,” “sadness,” “fear,” “anger,” “hate,” “surprise,” and “no emotion”) [2] were used. Hereafter, the emotion labels given by the 44 evaluators are referred to as L_m .

Table 1. Details of speakers and utterances

ID	Gender	#Utterance	ID	Gender	#Utterance	ID	Gender	#Utterance
sp1	Female	88	sp6	Female	23	sp11	Male	3
sp2	Male	66	sp7	Male	23	sp12	Male	3
sp3	Male	28	sp8	Male	14	sp13	Male	2
sp4	Male	27	sp9	Male	7	sp14	Male	1
sp5	Male	24	sp10	Female	5	sp15	Female	1

(sp6 is a child)

3.2 Supervised Signals

All utterances were labeled by the evaluators. However, an utterance was sometimes labeled with different emotions by the evaluators. Since EM_p needs the “correct” emotion for all training samples, we had to define the “correct” emotion from L_m .

In this experiment, we employed the “decision by majority” method. For each sample, the frequency was counted for each emotion, and the emotion with the maximum frequency was defined as the “correct” emotion. However, several samples could not be narrowed down to only one emotion because one or more different emotions had almost the same frequency as the maximum. There is no meaning if there are only one or two differences of frequency. Therefore, if the frequency of the second- or lower-ranked emotion was higher than 90 % of the maximum frequency, then that emotion was added to the “correct” emotions. In mathematical terms, all emotions that satisfied the equation $n_i > 0.9\hat{n}$ were selected as “correct” emotions. Note that n_i denotes the frequency of emotion i , and $\hat{n} = \max_i \{n_i\}$. As a result, there are one or more “correct” emotions for one utterance. The average number of emotions per utterance was 1.05. Hereafter, this “correct” emotion label is referred to as L_c .

To check the adequacy of this definition, the accordance ratio between L_c and L_m was calculated. The average recall rate was 67.9%, and the precision rate was 66.4%. On the other hand, the average accordance ratio between two labels in L_m was 54.5% (maximum 78.6%, minimum 29.6%). From this result, it becomes known that L_c is located at the “center” of L_m .

3.3 Calculation of Target Result

In the experiment described in section 3.1, the evaluators determined the L_m based on a diverse amount of information obtained from speech, facial expression, gesture, story line, and so on. However, the proposed method uses only the speech signals and thus is a more difficult task.

When an utterance is heard without any visual or contextual information, how accurately does a human recognize the speaker’s emotions? The level of accuracy in this case can be regarded as the upper limit of the performance of the proposed method. To calculate the upper limit, another evaluation experiment was carried

Table 2. Experimental results

Method	L_m			L_c		
	Recall	Precision	F-measure	Recall	Precision	F-measure
EM_p	53.5%	19.4%	28.5%	51.7%	23.4%	32.2%
EM_l	27.7%	27.7%	27.7%	31.5%	32.4%	31.9%
Merge	35.4%	30.2%	32.6%	44.5%	36.0%	39.8%

out. Utterance data was played back in random order to 15 evaluators, and they labeled the speaker’s emotion for all utterances. A subtitle corresponding to the utterance was shown simultaneously, but the utterances were only played once. The evaluators were not the same ones who made the L_m . Hereafter, the emotion label given by this experiment is referred to as L_s .

The average accordance ratio between L_s and L_m was 42.9%. We know that the average accordance ratio between two labels in L_m was 54.5%. The difference between 54.5% and 42.9% indicates the difficulty of emotion estimation from limited information. The average recall rate between L_s and L_c was 50.2% and the precision rate was 49.2%.

3.4 Prosodic Features

In this experiment, we used one of the standard prosodic feature sets [20]. This was defined for the emotion recognition competition, Emotion Challenge, at InterSPEECH 2009. It includes not only prosodic features but also spectrum features. In detail, it consists of zero-crossing rate, root mean square of frame power, pitch frequency, harmonics-to-noise ratio, and 12-dimensional MFCC. Delta coefficients (gradient of the sequence) were calculated for each parameter, and then 12 statistical parameters (average, standard deviation, maximum, minimum, range, etc.) were calculated for both the normal parameters and their delta coefficients. As a result, a 384-dimensional vector was calculated for one utterance. The openSMILE toolkit [43] was used for extraction.

3.5 Experimental Results

In this experiment, we could use only 315 utterances. Therefore, the “leave-one-speaker-out” method was employed. First, all samples uttered by sp1 were used for testing, and other samples were used for training. After that, the testing samples were changed to samples uttered by sp2, and other samples (including those uttered by sp1) were used for training. These steps were repeated 15 times, and the averaged recall and precision were calculated as the final result. By using this evaluation method, the final result can be regarded as the recognition performance for an unknown speaker’s utterance.

In the EM_p , the codebook size should be defined in advance. However, as seen in Table 1, the number of utterances differed greatly among the speakers. Therefore, the appropriate codebook size should be used for each speaker. In this

experiment, codebooks were constructed in several sizes (2^1 — 2^8 and the same as the number of training samples), and the best one was selected *a posteriori* for each speaker.

Table 2 shows the accordance ratio of each emotion recognition method and the emotion labels. From these results, the proposed method (“Merge”) showed the best performance of all. The performance of EM_p and EM_l was similar, but the integration step brought a dramatic improvement. In a comparison with the target result (described in Section 3.3), the experimental result reached 75.9% of the target result for L_m , and 80.1% of the target result for L_c . These results are very successful.

The final output of the proposed method included 1.17 emotions on average for one utterance. EM_p before integration output 4.22 emotions on average. Of course, this was not the best result (F-measure was 27.3%). However, integration with the labels given by EM_l requires a much greater number of emotions because the intersection of the two outputs should not be empty. Intersection results for 23.8% of samples were empty. The best result given by only EM_p included 2.76 emotions on average.

Figure 2 shows the histogram of F-measures between the results given by the proposed method and each label in L_m , which has 44 labels given by the 44 evaluators. It can be seen that many F-measures are located between 33%—39%. This means that several evaluators gave different (low F-measure) labels.

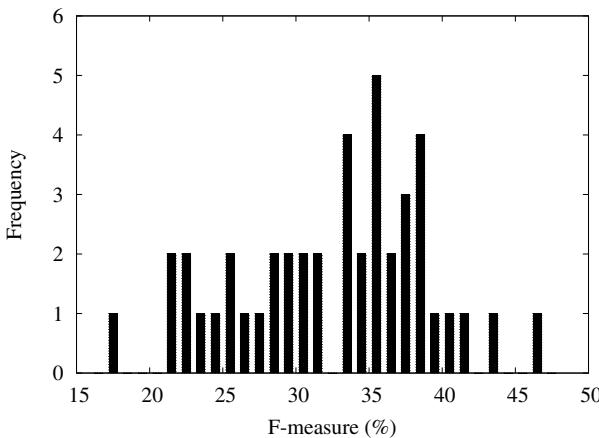


Fig. 2. Histogram of F-measures between results given by the proposed method and each labels in L_m

4 Conclusion

This paper describes a novel approach to emotion recognition based on both prosodic and linguistic features of speech. First, several prosodic features are extracted from the speaker’s utterance, and the clustering-based method is used to output possible emotions. Since different emotions in speech can have similar

prosodic features, the clustering-based method is employed. It makes clusters based only on the similarity of prosodic features, thus possible emotions can be output. At the same time, subtitles given by the speech recognizer are input to the linguistic-based emotion recognition agent, which is based on the “Association Mechanism.” Finally, the intersection of the two recognition results is integrated into the final result.

Experimental results showed that the proposed method achieved higher performance than either prosodic- or linguistic-based emotion recognition. In a comparison with manually labeled data, the proposed method gave 35.4% (recall), 30.2% (precision), and 32.6% (F-measure). On the other hand, human labeling of speech data gave 42.9%. This means that the proposed method performed at 75.9% in relation to human ability.

References

1. Cambridge University Engineering Department: Hidden Markov Model Toolkit, <http://htk.eng.cam.ac.uk/>
2. Ekman, P.: Argument for basic emotions. In: Cognition and Emotion, pp. 169–200 (1992)
3. Eyben, F., Wollmer, M., Schuller, B.: openSMILE — Speech and music interpretation by large-space extraction, <http://opensmile.sourceforge.net/>
4. Eyben, F., Wollmer, M., Schuller, B.: openEAR — Introducing the Munich open-source emotion and affect recognition toolkit. In: Proc. 4th International HU-MAINE Association Conference on Affective Computing and Intelligent Interaction 2009 (ACII 2009), vol. I, pp. 576–581 (2009)
5. Fujisaki, H.: Prosody, models, and spontaneous speech. In: Sagisaka, Y., Campbell, N., Higuhi, H. (eds.) Computing Prosody: Computational Models for Processing Spontaneous Speech, pp. 27–42. Springer, Heidelberg (1997)
6. Grimm, M., Kroschel, K., Mower, E., Narayanan, S.: Primitives-based evaluation and estimation of emotions in speech. *Speech Communication* 49(10-11), 787–800 (2007)
7. Hirose, T., Watabe, H., Kawaoka, T.: Automatic refinement method of concept-base considering the rule between concepts and frequency of appearance as an attribute. Technical Report of IEICE NLC2001-93, The institute of Electronics, Information and Communication Engineers (2002) (in Japanese)
8. Horiguchi, A., Tsuchiya, S., Kojima, K., Watabe, H., Kawaoka, T.: Constructing a sensuous judgment system based on conceptual processing. In: Gelbukh, A. (ed.) CICLING 2002. LNCS, vol. 2276, pp. 86–95. Springer, Heidelberg (2002)
9. Mera, K., Ichimura, T., Aizawa, T., Yamashita, T.: Invoking emotions in a dialog system based on word-impressions. *Transactions of the Japanese Society for Artificial Intelligence* 17(3), 186–195 (2002) (in Japanese)
10. Kojima, K., Watabe, H., Kawaoka, T.: A method of a concept-base construction for an association system: Deciding attribute weights based on the degree of attribute reliability. *Journal of Natural Language Processing* 9(5), 93–110 (2002) (in Japanese)
11. Lee, A., Kawahara, T., Shikano, K.: Julius — an open source real-time large vocabulary recognition engine. In: Proc. Eurospeech, pp. 1691–1694 (2001)
12. Lee, C.M., Narayanan, S.S., Pieraccini, R.: Combining acoustic and language information for emotion recognition. In: Proc. ICSLP 2002 (2002)

13. Linde, Y., Buzo, A., Gray, R.M.: An algorithm for vector quantizer design. *IEEE Trans. Communication* 28(1), 84–95 (1980)
14. Luengo, I., Navas, E., Hernaez, I.: Combining spectral and prosodic information for emotion recognition in the Interspeech 2009 Emotion Challenge. In: Proc. Interspeech, pp. 332–335 (2009)
15. Maekawa, K.: Corpus of spontaneous Japanese: Its design and evaluation. In: Proc. ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition, SSPR (2003)
16. Matsumoto, K., Mishina, K., Ren, F., Kuroiwa, S.: Emotion estimation algorithm based on emotion occurrence sentence pattern. *Journal of Natural Language Processing* 14(3), 239–271 (2007) (in Japanese)
17. Rigoll, G., Muller, R., Schuller, B.: Speech emotion recognition exploiting acoustic and linguistic information sources. In: Proc. SPECOM 2005, vol. 1, pp. 61–67 (2005)
18. Schuller, B., Muller, R., Lang, M., Rigoll, G.: Speaker independent emotion recognition by early fusion of acoustic and linguistic features within ensembles. In: Proc. Interspeech, pp. 805–808 (2005)
19. Schuller, B., Rigoll, G., Lang, M.: Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine — belief network architecture. In: Proc. ICASSP 2004, vol. 1, pp. 577–580 (2004)
20. Schuller, B., Steidl, S., Batliner, A.: The interspeech 2009 emotion challenge. In: Proc. Interspeech 2009, pp. 312–315 (2009)
21. Schuller, B., Villar, R.J., Rigoll, G., Lang, M.: Meta-classifiers in acoustic and linguistic feature fusion-based affect recognition. In: Proc. ICASSP 2005, vol. 1, pp. 325–329 (2005)
22. Tokuhisa, M., Okada, N.: A pattern comprehension approach to emotion arousal of an intelligent agent. *Transactions of Information Processing Society of Japan* 39(8), 2440–2451 (1998) (in Japanese)
23. Tsuchiya, S., Yoshimura, E., Ren, F., Watabe, H.: Emotion judgment based on relationship between speaker and sentential actor. In: Velásquez, J.D., Ríos, S.A., Howlett, R.J., Jain, L.C. (eds.) KES 2009. LNCS, vol. 5711, pp. 62–69. Springer, Heidelberg (2009)
24. Tsuchiya, S., Yoshimura, E., Watabe, H.: Emotion judgment method from an utterance sentence. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010. LNCS, vol. 6279, pp. 1–10. Springer, Heidelberg (2010)
25. Tsuchiya, S., Yoshimura, E., Watabe, H., Kawaoka, T.: The method of the emotion judgment based on an association mechanism. *Journal of Natural Language Processing* 14(3), 119–238 (2007) (in Japanese)
26. Watabe, H., Horiguchi, A., Kawaoka, T.: A sense retrieving method from a noun for the commonsense feeling judgment system. *Journal of Artificial Intelligence* 19(2), 73–82 (2004) (in Japanese)
27. Watabe, H., Kawaoka, T.: Measuring degree of association between concepts for commonsense judgments. *Journal of Natural Language Processing* 8(2), 39–54 (2001) (in Japanese)

Possibilistic Entropy: A New Method for Nonlinear Dynamical Analysis of Biosignals

Tuan D. Pham

School of Engineering and Information Technology,
The University of New South Wales,
Canberra, ACT 2600, Australia
t.pham@adfa.edu.au

Abstract. The theory of nonlinear dynamical systems has opened doors to discovering potential patterns hidden in complex time-series data. An attractive approach to nonlinear time-series analysis is the measure of predictability which characterizes the data in terms of entropy. A new entropy measure is presented in this paper as a new nonlinear dynamical method, which is based on the theory of possibility and the kriging computation. The proposed model has the potential for studying complex biosignals.

1 Introduction

Nonlinear dynamical analysis methods derived from the information theory for measuring the complexity of time-series data have been successfully applied to many scientific disciplines, including biology, physiology, medicine, biophysics, chemistry, and economics [1]. However, the impact of these methods has only been partly explored to date for a better understanding of physiological function [2]. Some important physiological findings based on the concepts of nonlinear dynamics were also addressed in [2], including four major methodology families: fractals, entropy measures, symbolic dynamics measures, and Poincaré plot representation. Among these four families, the entropy measures are the most widely used methods for studying biological and physiological time-series data.

The entropy approach is a powerful tool for understanding signal predictability or system complexity. The first method of this entropy family, known as approximate entropy (ApEn), was developed by Pincus [3]-[5]. ApEn is rooted in the work of Grassberger and Procaccia [6] and Eckmann and Ruelle [7], and widely applied in clinical cardiovascular studies and analysis of biomedical signals [8,9]. A low value of the approximate entropy indicates the time series is deterministic (low complexity); whereas a high value indicates the data is subject to randomness (high complexity) and therefore difficult to predict. In other words, lower entropy values indicate more regular the signals under study; whereas higher entropy values indicate more irregular the signals.

Extending the framework of approximate entropy (ApEn), sample entropy (SampEn) [10] and multiscale entropy (MSE) [11] were introduced to enhance the

predictability analysis of time-series data with particular reference to physiological signals. In general, both ApEn and SampEn estimate the probability that the sequences in a dataset which are initially closely related remain closely related, within a given tolerance, on the next incremental comparison. ApEn differs from SampEn in that its calculation involves counting a self-match for each sequence of a pattern, which leads to bias in ApEn [5]. SampEn is precisely the negative natural logarithm of the conditional probability that two sequences similar for m points remain similar at the next point, where self-matches are not included in calculation of the probability. Thus a lower value of SampEn also indicates more self-similarity in the time series. Based on the concept of fuzzy sets, a method named FuzzyEn was developed [12], where the similarity is defined by the degree of fuzziness and the shapes of the fuzzy membership functions.

This entropy measure family has been increasingly applied to many problems in biomedical engineering and other fields of life sciences [13][14]. However, it has been pointed out that ApEn suffers from two major drawbacks: 1) because it is a function of the length of the sequence under study, it yields entropy values lower than expected for short sequences, being due to the counting of a self-match for each sequence, which leads to bias [5]; 2) it can be inconsistent with different testing conditions using different parameters of the entropy index. SampEn does not count self-matches and therefore can reduce bias. It has been found that SampEn can provide better relative consistency than ApEn because it is largely independent of sequence length [10]. MSE measures complexity of time-series data by taking into account multiple time scales, but MSE uses SampEn to quantify the regularity of the data. Most recently, as another entropy method, namely GeoEntropy (GeoEn), has been developed [15]. Although GeoEn can relax the assumption of the parameter selections encountered by other entropy-based methods, it does not allow the continuous modeling of the similarity measure.

In this paper, possibilistic entropy for nonlinear analysis of time-series data is introduced. The proposed method have the capability of identifying the correlated structural (spatial) information of the data. This entropy measure is based on the notion of the theory of possibility [16], which is a fuzzy restriction acting as an elastic constraint on the values that may be assigned to the variable of similarity in our study. Its mechanism for similarity computation is carried out using the kriging estimator in geostatistics [17].

2 Possibilistic Entropy

We extend GeoEn to allow the modeling of similarity using the concept of possibility by firstly defining the experimental semi-variogram [17] of a sequence X , denoted by $\gamma(h)$, as

$$\gamma(h) = \frac{1}{2(n-h)} \sum_i^{n-h} (x_i - x_{i+h})^2 \quad (1)$$

where x_i is a value of X taken at location i , x_{i+h} another value taken at h distance away (for $h = 1$ in a time-series signal, every point is compared with its neighbors; and for $h = 2$, every point is compared with a point two spaces away), and n is the total number of points which gives $(n - h)$ as the total number of the pairs of points.

In geostatistics, the theoretical semi-variogram is defined as [17]

$$\gamma(h) = \begin{cases} s \left[1.5 \frac{h}{g} - 0.5 \left(\frac{h}{g} \right)^3 \right] & : h \leq g \\ s & : h > g \end{cases} \quad (2)$$

where g and s are called the *range* and the *sill* of the theoretical semi-variogram, respectively.

The geostatistical distance between two sub-sequences X_i and X_j of X can be defined by

$$d_{ij}(h) = |\gamma_{X_i}(h) - \gamma_{X_j}(h)| \quad (3)$$

Furthermore, the tolerance of a geostatistical self-similarity can be expressed as the absolute difference of the spatial variances of h and $h + 1$:

$$r_h = |\gamma_{X_i}(h + 1) - \gamma_{X_i}(h)| \quad (4)$$

We adopt the concept of signal error matching to derive a multiscale possibilistic entropy for handling time-series data by considering the following ordinary kriging system [17]:

$$\mathbf{C} \mathbf{a} = \mathbf{b} \quad (5)$$

where \mathbf{C} is the square and symmetrical matrix that represents the spatial covariances between the known signals, and \mathbf{b} is the vector that represents the spatial covariances between the unknown and known signals:

$$\mathbf{C} = \begin{bmatrix} \gamma_{11} & \cdots & \gamma_{1p} & 1 \\ \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdots & \cdot & \cdot \\ \gamma_{p1} & \cdots & \gamma_{pp} & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix}$$

where γ_{ij} is the semi-variance of x_i and x_j .

$$\mathbf{a} = [a_1 \cdots a_p - \lambda]^T$$

where a_k , $k = 1, \dots, p$ are called the kriging weights, and λ is a Lagrange multiplier.

$$\mathbf{b} = [\gamma_{n1} \cdots \gamma_{np} 1]^T$$

Thus the vector of the spatial predictor coefficients can be obtained by solving: $\mathbf{a} = \mathbf{C}^{-1} \mathbf{b}$.

The possibility of similarity between two sub-signals X_i and X_j of X using p points, denoted as $\mu_{ij}(p)$, can be defined in terms of the ratio of the kriging estimate errors:

$$\mu_{ij}(p) = \frac{\mathbf{a}_i^T \mathbf{b}_i}{\mathbf{a}_j^T \mathbf{b}_i} \quad (6)$$

where \mathbf{a}_i is defined in (5) which is the kriging prediction vector of X_i , \mathbf{b}_i is defined in (5) associated with X_i , and \mathbf{a}_j is the kriging prediction vector of X_j . It can be seen that the use of (6) allows a convenient way for processing long signals by considering only the prediction coefficients. The possibility is embedded in the ratio of the distortion in a continuous scale. When the two signals are identical, their degree of similarity has its maximum value of 1.

$$\omega_i^m(p) = \frac{1}{N - m - 1} \sum_{j=1, j \neq i}^{N-m} \mu_{ij}(p) \quad (7)$$

$$\phi^m(p) = \frac{1}{N - m} \sum_{i=1}^{N-m} \omega_i^m(p) \quad (8)$$

$$PossEn(p) = \ln \phi^m(p) - \ln \phi^{m+1}(p) \quad (9)$$

It is noted that there is a possibility that $PossEn(h)$ can be negative, and these negative values may not be convenient for the interpretation of multiscale possibilistic entropy. This negative affect is due to the fluctuations of the experimental semi-variograms at small values of the lag distance, particularly when h is between 1 and 3. This affect is well known to often cause the result of negative weights in kriging (geostatistical) estimates. A simple and effective strategy for correcting negative weights in kriging estimates was proposed by Journel and Rao [18]. This method determines the largest negative weight and adds an equivalent positive constant to all weights which are then normalized. Thus, negative values of the multiscale possibilistic entropy can be similarly handled using this strategy according to the following formulation:

$$PossEn(h)^* = \frac{PossEn(h) + \alpha}{\sum_h [PossEn(h) + \alpha]}, \quad \forall h \quad (10)$$

where $PossEn(h)^*$ is the corrected value of $PossEn(h)$, and α is defined as

$$\alpha = - \min_h PossEn(h) \quad (11)$$

3 Experiment

We are interested in applying the proposed possibilistic entropy methods for identifying cohorts of potential biomarkers from the mass spectrometry data of major adverse cardiac events (MACE); while other entropy measures are not able

to use the spatial information of the data to perform this task. The mass spectra used in this study are the same as those used in the original work described in [19], and were previously studied for the prediction of MACE [20] in which the identification of biomarkers was never investigated. The data are briefly described as follows. Two groups of plasma samples were used: MACE group of 60 patient samples: patients with chest pain and consistently negative Troponin T, but suffered MACE during the next period of 30 days or 6 months; and control group of 60 patient samples: patients with chest pain and consistently negative Troponin T and lived in next 5 years without any major cardiac events or death.

To increase the coverage of proteins in SELDI protein profiles, the blood samples were fractionated with HyperD Q (anion ion exchange) into six fractions. The protein profiles of fractions 1-6 were acquired with two SELDI Chips: IMAC and CM10. A total of 120 plasma samples, 24 reference samples, and 6 blanks were randomly divided into two groups (A and B) and were fractionated into six fractions using two 96-well plates containing anion exchange resin (Ciphergen, CA). Group A was processed in day 1 while Group B was processed on day 2. Two 96-well anion exchange resin plates were used to fractionate samples into six discrete fractions. All SELDI MS data were processed with CiphergenExpress 3.0 to generate peak maps. All spectra were pre-processed with the baseline subtraction and followed by normalization based on TIC with CiphergenExpress 3.0.

$PossEn(p)$ was applied to identify the subsequences of the samples spatially separated by lag h which constitute the most distinguishing characteristics in showing the difference between MACE and control profiles. This validation was carried by extracting from the original spectra all possible subsequences of samples separated by $h = 1, \dots, 20$. As a result, Figure 1 shows the average possibilistic entropy profiles of the MACE and control subspectra using $PossEn(p)$, where the value used for p is 6 (values of $p = 4, 6$, and 8 were used in the experiment and the three entropy profiles of MACE and control were found to be similar to each other, respectively; therefore all the profiles were not plotted to maintain the clarity of the graphical illustration). The possibilistic entropy profiles of the MACE and control subspectra generated by $PossEn(p)$ show a similar pattern to each other, in which the larger lag h the higher the entropy value indicating the more complexity of the signals. The MACE subspectra have higher possibility values than the control. It can be easily observed that Figure 1 shows the largest gap between the profiles at $h=16$. These results suggest that the panels of the protein peaks spatially separated by such a lag distance have the best ability for discerning the difference between MACE and control.

To further validate the finding of potential panels of the biomarkers, we performed the classification of MACE and control subspectra using the following rule: assign subspectra i to class c_k (MACE or control) $\Leftrightarrow d_{ij} = \min_j(d_{ij}), j \in c_k \forall k, i \neq j$, where $d_{ij} = (\mathbf{a}_i^T \mathbf{b}_j) / (\mathbf{a}_j^T \mathbf{b}_i)$; in which \mathbf{a}_i is defined in (5) which is the kriging prediction vector of subspectra i , \mathbf{b}_i is defined in (5) associated with i , and \mathbf{a}_j is the kriging prediction vector of subspectra j .

The leave-one-out method was then applied to carry out the classification task for each group of the h -spaced subspectra of MACE and control. The average

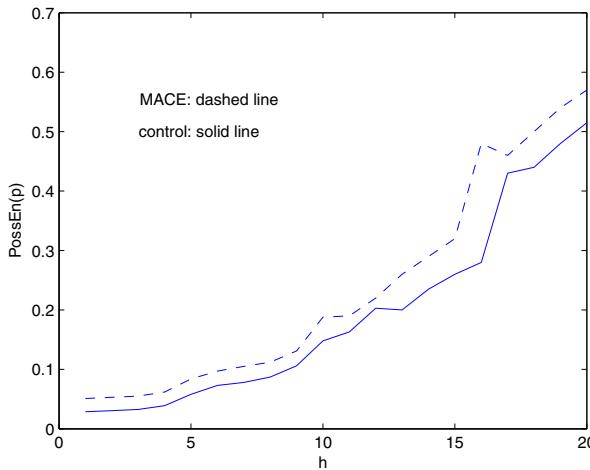


Fig. 1. Possibilistic entropy profiles of MACE and control using $PossEn(p)$

Table 1. Average (subscript *a*) and best (subscript *b*) sensitivity (SEN) and specificity (SPE) using subspectra of h -spaced samples of MACE and control

h	SEN_a (%)	SPE_a (%)	SEN_b (%)	SPE_b (%)
2	64.11	65.47	65.29	66.15
4	64.24	65.52	65.10	66.76
6	64.38	65.22	65.72	66.50
8	65.68	66.89	66.71	67.59
10	69.06	68.11	72.11	71.44
12	66.30	65.87	67.48	66.52
14	68.19	67.08	70.21	68.54
16	76.37	74.19	86.27	83.39
18	67.05	65.71	68.67	66.81
20	67.48	67.39	68.91	68.60

classification results in terms of sensitivity and specificity are shown in Table I for different lag-spaced subspectra, where the values of h are selected as the multiples of 2; and $p = 6$ as the length of the kriging prediction vectors. Sensitivity is the percentage of MACE samples that are correctly identified, whereas specificity is the percentage of control samples that are correctly identified. The use of the subspectra taken at $h = 16$ gives the highest classification rates in both sensitivity. The classification using the subspectra consisting the samples at $h = 16$ has an average sensitivity of 7% better than the second best at $h=10$, and 12% better than the lowest at $h = 2$; and an average specificity of about 6% better than the second best at $h=10$, and 9% better than the lowest at $h = 2$.

As for the best sensitivity comparison, the use of the best subset of the subspectra consisting of the samples at $h = 16$ outperforms about 14% better than

the second best at $h=10$, and about 21% better than the lowest at $h = 2$. Regarding the best specificity comparison, the use of the best subset of the subspectra gives about 12% better than the second best at $h=10$, and about 17% better than the lowest at $h = 2$. Thus, the classification results, with particular reference to the best subset of the subspectra, confirm the suggestion that the protein peaks taken at intervals of lag space $h = 16$ are potential biomarkers, which have the best power to predict the disease. As another observation of the results shown in Table 1, both sensitivity and specificity of the subspectra of $h = 16$ have the largest difference between the average and best values among other h -spaced subspectra. This difference can be intuitively interpreted in that although the subspectra of $h=16$ are suggested to be panels of potential biomarkers, only a few are of the biomarkers.

4 Conclusion

A new method for nonlinear analysis of time-series data has been discussed in the foregoing sections. The incorporation of the concept of possibility derived from the theory of fuzzy sets allows a continuous and natural modeling of pattern similarity, where the difference between the patterns is subtle to be captured by any hard-threshold measures. The kriging computation allows an efficient mechanism for the callation of similarity in terms of its signal matching scheme. The proposed method was applied for identifying panels of potential biomarkers from mass spectrometry data for early prediction of major adverse cardiac events. The experimental results have shown the potential application of the probabilistic entropy, which can be useful for handling many other types of biosignals.

References

1. Kaplan, D., Glass, L.: Understanding Nonlinear Dynamics. Springer, New York (1995)
2. Voss, A., et al.: Methods derived from nonlinear dynamics for analysing heart rate variability. *Phil. Trans. R. Soc. A* 367, 277–296 (2009)
3. Pincus, S.M.: Approximate entropy as a measure of system complexity. *Proc. Natl. Acad. Sci. USA* 88, 2297–2301 (1991)
4. Pincus, S.M., Goldberger, A.L.: Physiological time-series analysis: what does regularity quantify? *Am. J. Physiol.* 4, H1643–H1656 (1994)
5. Pincus, S.M.: Approximate entropy (ApEn) as a complexity measure. *Chaos* 5, 110–117 (1995)
6. Grassberger, P., Procaccia, I.: Estimation of the Kolmogorov entropy from a chaotic signal. *Phys. Rev. A* 28, 2591–2593 (1983)
7. Eckmann, J.P., Ruelle, D.: Ergodic theory of chaos and strange attractors. *Rev. Modern. Phys.* 57, 617–654 (1985)
8. Kannathal, N., et al.: Entropies for detection of epilepsy in EEG. *Comput. Meth. Programs Biomed.* 80, 187–194 (2005)

9. Srinivasan, V., Eswaran, C., Sriraam, N.: Approximate entropy-based epileptic EEG detection using artificial neural networks. *IEEE Trans Information Technology in Biomedicine* 11, 288–295 (2007)
10. Richman, J.S., Moorman, J.R.: Physiological time-series analysis using approximate entropy and sample entropy. *Amer. J. Physiol. Heart Circ. Physiol.* 278, H2039–H2049 (2000)
11. Costa, M., Goldberger, A.L., Peng, C.K.: Multiscale entropy analysis of complex physiologic time series. *Phys. Rev. Lett.* 89, 068102-1–068102-4 (2002)
12. Chen, W., Zhuang, J., Yu, W., Wang, Z.: Measuring complexity using FuzzyEn, ApEn, and SampEn. *Medical Engineering & Physics* 31, 61–68 (2009)
13. Lewis, M.J., Short, A.L.: Sample entropy of electrocardiographic RR and QT time-series data during rest and exercise. *Physiological Measurement* 28, 731–744 (2007)
14. Lee, M.-Y., Yang, C.-S.: Entropy-based feature extraction and decision tree induction for breast cancer diagnosis with standardized thermograph images. *Comput. Meth. Programs Biomed.* 100, 269–282 (2010)
15. Pham, T.D.: GeoEntropy: a measure of complexity and similarity. *Pattern Recognition* 43, 887–896 (2010)
16. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets Syst.* 100, 9–34 (1999)
17. Isaaks, E.H., Srivastava, R.M.: An Introduction to Applied Geostatistics. Oxford University Press, New York (1989)
18. Journel, A.G., Rao, S.E.: Deriving conditional distribution from ordinary kriging, Stanford Center for Reservoir Forcasting. Stanford University Report (29), 25 (1996)
19. Brennan, M.-L., Penn, M.S., Van Lente, F., Nambi, V., Shishehbor, M.H., Aviles, R.J., Goormastic, M., Pepoy, M.L., McErlean, E.S., Topol, E.J., Nissen, S.E., Hazen, S.L.: Prognostic value of myeloperoxidase in patients with chest pain. *New Eng. J. Med.* 348, 1595–1604 (2003)
20. Pham, T.D., Wang, H., Zhou, X., Beck, D., Brandl, M., Hoehn, G., Azok, J., Brennan, M.L., Hazen, S.L., Li, K., Wong, S.T.C.: Computational prediction models for early detection of risk of cardiovascular events using mass spectrometry data. *IEEE Trans Information Technology in Biomedicine* 12, 636–643 (2008)

Necessary Tools Choice in a Particular Situation for Computer Conversation

Eriko Yoshimura¹, Misako Imono², Seiji Tsuchiya¹, and Hirokazu Watabe¹

¹ Faculty of Science and Engineering, Doshisha University,
Kyo-Tanabe, Kyoto, 610-0394, Japan

² Graduate School of Engineering, Doshisha University,
Kyo-Tanabe, Kyoto, 610-0394, Japan

{eyoshimura,mimono,stsuchiya,watabe}@indy.doshisha.ac.jp

Abstract. Human beings make associations based on their general knowledge, with intellectual thought activity forming the basis of natural conversation. When human beings converse, they first absorb what the other party is saying, then use their analogical and associative abilities to continue the conversation based on naturally acquired general knowledge. Such general knowledge-based conversation is difficult to simulate using the language data and methods that have been acquired thus far. Therefore, it is necessary to provide machines with the intellectual structure they require to understand the semantic connections between words, determine how these words relate to one another, and render sensible judgments. In this paper, we propose a method for constructing a system aimed at determining what tools are necessary in a particular situation based on the intellectual structure of sensible judgment. For example, this system would examine data input like “cut the cabbage” as it appears in a conversational sentence, and associate it with relevant tools such as “kitchen knife” and “cutting board”. In this manner, rather than by simply parsing the phrase as linguistic data, the system allows for responsive dialogue that is relevant to the situation.

Keywords: Necessary Tools Choice, Computer conversation, Natural languages.

1 Introduction

In recent years, electronic devices and various other machines have become increasingly sophisticated and intelligent. The shared future vision for these machines is for them to coexist seamlessly with human beings. Advancements toward this goal include the development of numerous robots, some of which are capable of walking on two legs, running, and even dancing [1][2]. Through such developments, machine forms are being created that appear increasingly humanlike. At this stage, in order to seamlessly and efficiently coexist with humanity, such machines have an increasing need for intelligence and the capacity to converse naturally with human beings. In the future, the ability to engage in conversation with human beings will be indispensable

[3] [4] [5]. As a result, the field of natural language processing has garnered significant attention recently.

Dramatic advances have also been made in computer memory capacity and processing speed. Alongside such advances, engineers in the field of natural language processing are working to compile a large database of linguistic knowledge. In the context of this study, the term “language” encompasses a data structure that includes grammar, synonyms, equivalent terms, antonyms and other related factors. There is great significance in incorporating such knowledge into a comprehensive database [6] [7]. Typically, however, human conversation is not simply linguistic data. It consists of words joined together by intelligent thought activities that include intentions and nuance. This intellectual activity is difficult to emulate by simply compiling a large linguistic database from recorded conversations [8]. Yet it is exactly this kind of intelligent thought activity that allows human beings to make sense of and understand a conversation.

Human beings make associations based on their general knowledge, with intellectual thought activity forming the basis of natural conversation. When human beings converse, they first absorb what the other party is saying, then use their analogical and associative abilities to continue the conversation based on naturally acquired general knowledge. Such general knowledge-based conversation is difficult to simulate using the language data and methods that have been acquired thus far. Therefore, it is necessary to provide machines with the intellectual structure they require to understand the semantic connections between words, determine how these words relate to one another, and render sensible judgments.

In this paper, we propose a method for constructing a system aimed at determining what tools are necessary in a particular situation based on the intellectual structure of sensible judgment. For example, this system would examine data input like “cut the cabbage” as it appears in a conversational sentence, and associate it with relevant tools such as “kitchen knife” and “cutting board”. In this manner, rather than by simply parsing the phrase as linguistic data, the system allows for responsive dialogue that is relevant to the situation.

2 The Concept Association^{[9][10]}

Under normal conversational circumstances, human beings have the innate ability to appropriately interpret any linguistic information received. This is possible because they have accumulated basic linguistic knowledge and understand word concepts based on their experiences. In other words, the ability to recall concepts related to a certain word plays a vital role in conversation. To be able to sensibly evaluate a word, listeners must understand basic information about it. Accordingly, we intend to model human knowledge of conversations and words into a methodology that can be used to instruct machines. We believe that by doing so, we can create a conversational structure resembling human conversation.

However, we first must determine the semantic similarity between words in order to allow machines to process them in a more humanlike fashion. This will permit machines to determine the semantic relationships between words with a level of intuition close to that of human beings. For example, machines should be capable of

evaluating word pairs like "woman-lady" or "mountain-hill" as synonymous (or at least similar), word pairs such as "mountain-river" and "sunset-red" as closely related, and pairs like "mountain-desk" and "train-sky" as normally unrelated.

The Concept Association Mechanism incorporates word-to-word relationships as common knowledge. This is a structure for capturing various word relationships. This section describes the Concept Base and the Degree of Association.

The Concept Base is a knowledge base consisting of words (concepts) and word clusters (attributes) that express the meaning of these words. This is automatically constructed from multiple sources, such as Japanese dictionaries and contains approximately 120,000 registered words organized in sets of concepts and attributes. An arbitrary concept, A , is defined as a cluster of paired values, consisting of attribute, a_i , which expresses the meaning and features of the concept, and weight, w_i , which expresses the importance of attribute a_i , in expressing concept A :

$$A = \{(a_1, w_1), (a_2, w_2), \dots, (a_N, w_N)\}$$

Attribute a_i is called the first-order attribute of concept A . In turn, an attribute of a_i (taking a_i as a concept) is called a second-order attribute of concept A .

	train, 0.36	railroad, 0.10	a_i, w_i	Primary Attributes
train	train, 0.36	railroad, 0.10	\dots	a_{ij}, w_{ij}
	railroad, 0.10	subway, 0.25	\dots	a_{i2}, w_{i2}
	\vdots	\vdots	\vdots	\vdots
	a_{1j}, w_{1j}	a_{2j}, w_{2j}	\dots	a_{ij}, w_{ij}

Secondary Attributes

Fig. 1. Example demonstrating the Concept "train" expanded as far as Secondary Attributes

Because Primary Attributes a_i of concept A are taken as the concepts defined in the Concept Base, attributes can be similarly elucidated from a_i . The Attributes a_{ij} of a_i are called Secondary Attributes of concept A . Figure 1 shows the elements of the Concept "train" expanded as far as the Secondary Attributes. The method for calculating the Degree of Association involves developing each concept up to second-order attributes, determining the optimum combination of first-order attributes by a process of calculation using weights, and evaluating the number of these matching attributes.

For Concepts A and B with Primary Attributes a_i and b_j and Weights u_i and v_j , if the numbers of attributes are L and M , respectively ($L \leq M$), the concepts can be expressed as follows:

$$A = \{(a_1, u_1), (a_2, u_2), \dots, (a_L, u_L)\}$$

$$B = \{(b_1, v_1), (b_2, v_2), \dots, (b_M, v_M)\}$$

The Degree of Identity (A, B) between Concepts A and B is defined as follows (the sum of the weights of the various concepts is normalized to 1):

$$I(A, B) = \sum_{a_i=b_j} \min(u_i, v_j) \quad (1)$$

The method means available degree is common weight of common attribute. So based this idea, the degree of identity defines min of two weights. All attributes of two concepts get most appropriate partner using this degree of identify. The degree of association is calculated by using the degree of identify of corresponding attribute and corrected value. (Refer to [9][10])

The value of the Degree of Association is a real number between 0 and 1. The higher the number is, the higher the association of the word. Table 1 lists examples of the degree of association.

Table 1. Examples of the degree of association

Concept A	Concept B	Degree of association between A and B
Flower	Cherry blossom	0.208
Flower	Car	0.0008
Car	Bicycle	0.23

3 Necessary Tools Choice in a Particular Situation

The target input phrase is a verb and noun (object) set that “needs” a tool. Examples of such phrases include “catch a fish”, “hang clothes”, “hit a ball”, etc.

In the first step, the system searches Internet to extract nouns that are likely to occur with the verb. Accepting these nouns as candidate words, the system then uses the NTT thesaurus [11] to refine the search towards determining possible tools. Finally, it checks the relationship of the noun with the input noun using concept association, and outputs potentially appropriate associated words. Figure 2 shows the flow of the process.

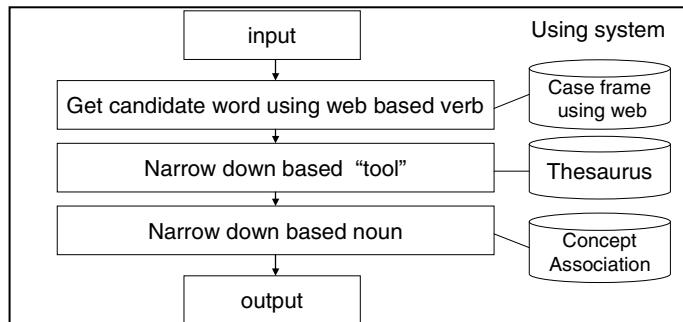


Fig. 2. System flow

3.1 Extracting Candidate Words Using the Large Internet Case Frame ^[12]

The Large Internet Case Frame (hereinafter referred to as "Case Frame System") is a knowledge base that is automatically constructed from the approximately 500 million sentences of Japanese text online and comprising approximately 90,000 verb. The system evaluates declinable words and their situations (what particles are used), and then obtains nouns related to the declinable words from online sources based on usage

frequency. Since we constructed the Needed Tool Judgment System described in this paper using the Japanese language, the Case Frame System is also Japanese-based. Furthermore, because the Case Frame System extracts everyday words for each usage online, it can collect a wide range of associated words and is not limited to tools.

For this paper, we selected candidate nouns using declinable words and the “de”-case in order to propose necessary tools for the input phrase. The Japanese language “de”-case refers to a noun complement that is connected to and expresses the relation with the predicate using the case-marking particle “de”. “De”-case nouns can express “locations”, “reasons”, “tools”, “methods” or “materials” for the predicate. Table 2 provides examples of expressions with “de”-case nouns.

Table 2. “De”-Case Noun Expressions

“De”-Case Expressions	Example
Location	<i>Tomodachi no ie de asobu</i> “We will play at a friend’s house ”
Reason	<i>Kaze de yasunda</i> “I had some time off because of my cold ”
Method	<i>Basu de iku</i> “I will go by bus ”
Material	<i>Sumi de yaku</i> “Burn it with charcoal ”
Tool	<i>Houcho de kiru</i> “Cut it with a kitchen knife ”

As shown in Table 2, tools are one of the noun types that can be extracted from phrases using declinable words and the “de”-case.

Using the Case Frame System for the input phrase *ki wo kiru* (“cut wood”) with the declinable word *kiru* (“to cut”) and the “de”-case, we can obtain candidate words such as “time, kitchen knife, scissors, knife, box cutter, saw, katana, up, hand, sword, violation, scalpel, swift attack, shape, length, chainsaw, hairdresser, shoulder and barber”.

3.2 Refining Candidate Words Down to Tool Choices

From the wide range of candidate words obtained online, we now extract the tool choices. Specifically, we used the NTT thesaurus to search for candidate words that express the concept of tools.

The NTT thesaurus uses a tree structure to show super-sub and part-whole relations for 2,710 semantic attributes (nodes) expressing the semantic usage of general nouns. Approximately 130,000 leaf words are registered as nouns at each node. Words in sub nodes inherit the semantic attributes of their parent node. The NTT thesaurus has a “Tool” node. Nouns with the Tool node as its superior nodes can be said to have inherited “tool” features. Thus, we can refine our candidate words down to tool choices by extracting nouns that have the Tool node. Figure 3 shows an overview of a section of the NTT thesaurus.

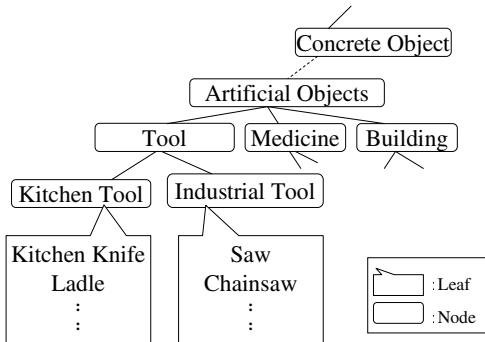


Fig. 3. NTT thesaurus section overview

In Section 3.1, we obtained the candidate words “time, kitchen knife, scissors, knife, box cutter, saw, katana, up, hand, sword, violation, scalpel, swift attack, shape, length, chainsaw, hairdresser, shoulder and barber” for the input phrase “ki wo kiru” (“cut wood”). Using the above method to refine these candidate words yields the following: “kitchen knife, scissors, knife, box cutter, saw, katana, sword, scalpel and chainsaw”.

3.3 Refining Candidate Words Based on Relation to the Input Phrase Noun

The method used in Section 3.2 yielded “kitchen knife, scissors, knife, box cutter, saw, katana, sword, scalpel and chainsaw” as cutting tools from the input phrase “ki wo kiru” (“cut wood”). However, because this group does not take into account the relation to the input phrase’s noun, the collected options include “kitchen knife, scissors, knife, box cutter, katana, sword and scalpel,” none of which are appropriate tools for cutting wood. Therefore, we must now refine the candidate words based on their relation to the input phrase noun. To accomplish this, we quantified the relevance of the candidate words to the input noun by using a relevance calculation. The system then determines relevance by extracting relevant words that exceed a set threshold, resulting in the output of “saw” and “chainsaw” as tools suitable for cutting wood.

4 Evaluation

We obtained the phrases used in our evaluation by requesting five candidate phrases from 26 test subjects. These candidate phrases were to consist of a verb and noun (object) set that needed a tool. After eliminating duplicates from the obtained data, we selected 100 candidate phrases for use as evaluation data.

After processing the 100 evaluation phrases through our system, we then had three test subjects evaluate the system output to determine whether the chosen candidate words were appropriate.

For the overall evaluation results, all output candidate words were evaluated as appropriate (O), sensible (N), or nonsensical (X). These results are explained below. For this paper, the total of (O) and (N) results were tabulated to provide verification of the system's precision.

O: Two or three test subjects deemed the candidate to be appropriate

N: One test subject deemed the candidate to be appropriate

X: All three test subjects deemed the candidate to be inappropriate

4.1 Output Word Evaluation

A total of 273 words were output from the 100 evaluation phrases, indicating that an average of 2.73 words was output per phrase. The evaluation results for the 273 output words were as follows: (O): 48.71%, (N): 22.71%, (X): 28.57%. This indicates the precision level of our system is 71.42%.

The total output word evaluation results are shown in Fig. 4 below:

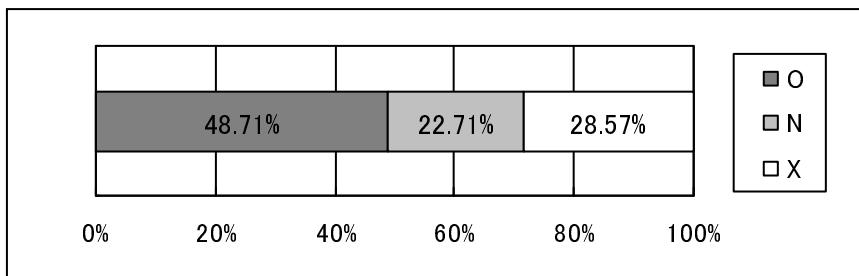


Fig. 4. Output word evaluation results

4.2 Evaluation of Output Words by Phrase

The averages for the evaluation method described in Section 4.1 cannot be used for each input phrase if any evaluation phrases result in multiple output words judged appropriate (O). Therefore, we evaluated the output for each phrase based on whether it includes output words receiving an (O) or (N):

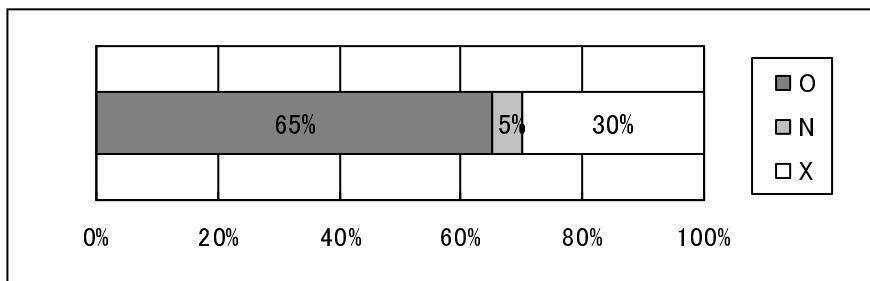
O: At least one output word was deemed appropriate (O)

N: No output words were deemed appropriate (O),

at least one word was deemed sensible (N)

X: No output words were deemed appropriate (O) or sensible (N)

Using this standard, the following results were obtained for the 100 phrases evaluated: (O): 65, (N): 5, (X): 30. These results indicate that system precision per phrase was 70%. Evaluation results for output word by phrase are shown in Fig. 5 below:

**Fig. 5.** Output word by phrase evaluation results

4.3 Considerations

System output examples shown in Table 3 below:

Table 3. Output Example

	Input	Output	Eval.
1	<i>Cha wo wakasu</i> ("Boil tea")	Pot	O
		Kettle	O
		Iron kettle	O
		Tea kettle	O
2	<i>Tsuchi wo horu</i> ("Dig in the dirt")	Spade	O
		Shovel	O
		Pickaxe	O
		Hoe	O
		Ground	X
3	<i>Sushi wo taberu</i> ("Eat sushi")	Soy sauce	O
		Vinegar	N
4	<i>Enpitsu wo kezuru</i> ("Sharpen a pencil")	Knife	O
		Rasp	X
		Blade	N
		File	X
		Shaving plane	X
5	<i>Sakana wo kau</i> ("Keep a fish as a pet")	Cage	X

As mentioned above, system precision was 71.42% for output words and 70% for output words by phrase. As these precision values are almost equal, we could conclude that many of the failed output words were for phrases that did not garner any correct output words.

As for why no correct output words were found for some phrases, there are two possibilities. First, no appropriate word may have been included in the candidate words obtained. Second, an associated tool might have been included at the candidate

stage that was dropped when the search was refined. Item 4 in Table 3 provided an example of this result from the phrase, “sharpen a pencil.” The term “pencil sharpener” was initially chosen as a candidate, but it was later eliminated because it is a compound term consisting of “pencil” and “sharpener,” and its relevance was not calculated properly. Despite this, with a correct extraction rate of 71.42%, we believe this method can be effective. However, it will be necessary to improve its precision by developing a more appropriate calculation method.

5 Conclusion

In this paper, we discussed a method for providing machines with the general knowledge needed to take semantic structure into account when engaged in conversation, instead of merely treating phrases as data. In particular, we showed that machines could correctly associate the tools needed for certain actions, and proposed a method that would provide them with general tools knowledge. More specifically, we proposed a method for constructing a system that could help determine what tools are necessary in a particular situation based on the intellectual structure of sensible judgment.

Our proposed method resulted in the creation of a system that had a 71.42% precision rate for output words. When used, the system does not simply parse the conversational sentence as linguistic data; it takes actions that make possible responsive dialogue between human beings and machines.

Acknowledgment. This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (Young Scientists (B), 21700241).

References

1. [Sony] Sony, <http://www.sony.jp/products/Consumer/aibo/> (accessed 2011-02-01)
2. [Honda] Honda, <http://www.honda.co.jp/ASIMO/> (accessed 2011-02-01)
3. Weizenbaum, J.: ELIZA - A Computer Program For the Study of Natural Language Communication Between Man and Machine. Communications of the Association For Computing Machinery 9(1), 36–45 (1965)
4. Richard, R.S., Wallace, S.: Alicebot, <http://alicebot.blogspot.com/> (accesssed 2011-02-01)
5. Carpenter, R.: Jabberwacky – live chat bot –AL artificial intelligence chatbot – jabber wacky – talking robot – chatbots – chatterbot – chatterbots – javverwocky – take a Turning Test Loebner Prize – Chatterbox C, <http://www.jabberwacky.com/> (accessed 2011-02-01)
6. Sekine, S., Inui, K., Torisawa, K.: Corpus Based Knowledge Engineering, <http://nlp.cs.nyu.edu/sekine/CBKE/> (accessed 2011-02-01)
7. Kurohashi, S., Nagao, M.: A method of case structure analysis for Japanese sentences based on examples in case frame dictionary. IEICE Transactions on Information and Systems E77-D(2), 227–239 (1994)

8. Yoshimura, E., Watabe, H., Kawaoka, T.: An Automatic Enhancing Method of Greeting Sentences Using Association Knowledge Mechanism. *Journal of Natural Language Processing* 13(1), 117–141 (2006)
9. Watabe, H., Kawaoka, T.: The Degree of Association between Concepts using the Chain of Concepts. In: Proc. of SM 2001, pp. 877–881 (2001)
10. Okumura, N., Yoshimura, E., Watabe, H., Kawaoka, T.: An Association Method Using Concept-Base. In: Apolloni, B., Howlett, R.J., Jain, L. (eds.) *KES 2007, Part I. LNCS (LNAI)*, vol. 4692, pp. 604–611. Springer, Heidelberg (2007)
11. NTT Communication Science Laboratory, NTT Thesaurus, *NIHONGOGOITAIKEI* (Iwanami Shoten book, 1997)
12. Kawahara, D., Kurohashi, S.: Case Frame Compilation from the Web using High-Performance Computing. In: Proceedings of the 5th International Conference on Language Resources and Evaluation, pp. 1344–1347 (2006)

Recommendation and Diagnosis Services with Structure Analysis of Presentation Documents

Shinobu Hasegawa¹, Akihide Tanida², and Akihiro Kashihara²

¹ Japan Advanced Institute of Science and Technology,

1-1, Asahidai, Nomi, Ishikawa, 923-1292, Japan

² The University of Electro-Communications,

1-5-1, Chofugaoka, Chofu, Tokyo, 182-8585 Japan

hasegawa@jaist.ac.jp, {tanida,kashihara}@ice.uec.ac.jp

Abstract. The main topic addressed in this paper is how to help novice researchers compose and improve their presentation documents by means of presentation heuristics shared by the members in a laboratory. The key idea is to propose a framework of presentation structure, which represents semantic roles of and relations among presentation slides included in the documents with metadata. Following the framework, this paper introduces a machine learning technique for automatically analyzing a typical presentation structure as presentation heuristics from the repository of the documents accumulated in the laboratory. This paper also demonstrates interactive Web services that recommend the metadata to be attached to the documents newly composed, and that diagnose the presentation structure of the documents by comparison with the presentation heuristics.

Keywords: presentation structure, presentation heuristics, structure analysis, metadata recommendation, structure diagnosis.

1 Introduction

In daily research activities, a large volume of contents is created and used by researchers and students as novice researchers in a laboratory. In particular, composing presentation documents is one of the most important activities so that they can publish the research findings with well-organized representation. It also involves constructing a semantic structure specifying what to present and what order to present [1]. We call it presentation structure. However, it is quite difficult for novice researchers to construct the presentation structure since they have few experiences of composing presentation documents. They also have fewer heuristics necessary for the document composition, which is often shared by the laboratory members. The main goal of this paper is to help the novices compose and improve their presentation documents on their own by means of presentation heuristics to be extracted from the presentation documents accumulated by the laboratory members. The presentation heuristics is represented as typical presentation structure.

A presentation document generally includes a number of slides, which have their respective semantic roles and semantic relationships among them. Such semantic

information forms the presentation structure. In order to extract the presentation heuristics, it is essential to refer to the presentation structure. However, it is often embedded in the documents since it would be time-consuming and difficult for the novices to extract suitable semantic information from the slides.

The main issue addressed in this paper is how to effectively help the novices construct the presentation structure of the presentation document they are newly composing. Our approach to this issue is to provide the novices with interactive Web services that recommend the metadata to be attached to the document newly composed, and that diagnose the presentation structure of the document. Towards these services, this paper proposes a framework of the presentation structure, which represents semantic information included in presentation documents with metadata [2]. Following this framework, we also utilize a machine learning technique to analyze a typical presentation structure as presentation heuristics shared by the laboratory members from the repository of the documents attached with the metadata in advance, which are accumulated in our laboratory. Such analysis allows the recommendation and diagnosis services. These services could help the researchers and students become aware of the typical presentation structure that are shared and followed by the laboratory members. Such awareness would contribute to developing skills in composing presentation documents and representing the research findings in more suitable way as a part of laboratory education.

2 Presentation Documents with Metadata

2.1 Presentation Composition Task

Presentation is an important research activity for brushing up the research itself in laboratory meeting and for publishing research findings in international/national conferences. It needs deciding what to present and what order to present to compose the presentation documents including a number of slides, which involves designing the contents of the slides and constructing the presentation structure. Although oral presentation is important for making good presentation, this paper focuses on constructing the presentation structure since it would contribute to developing skills in logically thinking and representing their research findings.

In order to compose a proper presentation document, it is necessary to represent not only research findings but also the presentation structure. Actually, good presentation documents are viewed as a knowledge repository of research since they have well-organized presentation structure that consists of several semantic roles of the slides such as "Background", "Purpose", "System", "Experiment", and "Conclusion" of the research. Furthermore, the presentation structure is often followed by heuristics peculiar to the laboratory. However, such heuristics is embedded in the presentation documents. It is accordingly difficult task for the novice researchers to give attention to the presentation heuristics used for composing and improving the presentation documents. In addition, it is not so easy for the novices to construct the presentation structure of the presentation documents they compose.

In general, expert researchers are not always good teachers for presentation. Of course, they could point out and fix inappropriateness of the presentation documents composed by the novices. But, it is not easy to teach skills in composing the

presentation documents directly to the novices. In addition, appropriate presentation structure would depend on diverse factors, such as presentation time limitation, philosophy in the laboratory, audiences, and research domain, which could be acquired heuristically through the daily research activities. In order to resolve these problems, we first propose a presentation structure framework.

2.2 Presentation Structure Framework

In this paper, the presentation structure framework provides a metadata model for representing presentation structures embedded in the presentation documents [3]. The presentation structure implies some heuristics for composing the documents, which could be shared by the laboratory members. In order to represent the presentation structure explicitly, this framework provides four types of metadata as shown in Fig. 1.

Slide metadata represent the semantic roles of each slide included in a presentation document, which not necessarily correspond to the title of each slide. Segment metadata also represent a sequence of the slide metadata in the document. We have defined four kinds of segment metadata each of which includes related slide metadata. Relation metadata represent sequential or hierarchical relationships among the slide metadata and segment metadata. File metadata represent some attributes of the document. Fig. 1 shows typical examples of metadata in our laboratory.

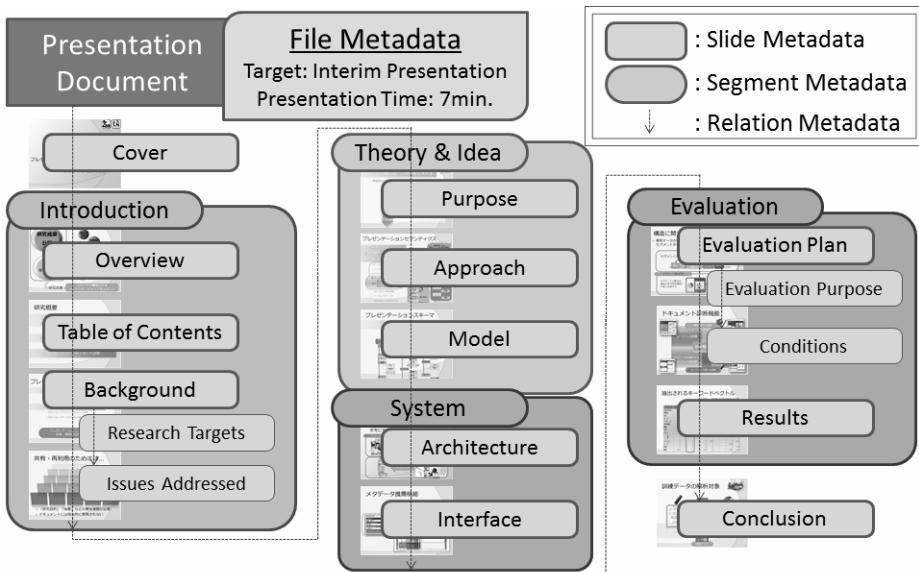


Fig. 1. Overview of Presentation Structure Framework

2.3 Approaches

Attaching the metadata to the documents is a time-consuming and complicated task as follows: (1) some metadata expressions are often taken different among researchers in their own ways even granted they have the same meaning, (2) it is difficult to detect

the slide metadata from its contents, and to also detect the segment and relation metadata from the slide sequence.

In order to resolve these difficulties, it is required to recommend the presentation structure as the metadata. Towards the metadata recommendation, we have utilized a machine learning technique to conduct structure analysis of the presentation documents accumulated in our laboratory members, which are attached with metadata in advance. As the results of the analysis, we have obtained the relationships between the presentation slides and its metadata, and the relationships among them as the structure information. The machine learning technique has also detected a schema of presentation structure that represents semantic features tendency among the accumulated documents. We call it presentation schema, which represents presentation heuristics shared by the laboratory members. The presentation schema enables the laboratory members to become aware of the gaps of the presentation structure between their documents and the schema. Such awareness is increased by the diagnosis service.

2.4 Related Work

This section briefly describes related work on presentation support from semantic information and technical points of view. Kohlhase [4] developed CPoint as a semantic PowerPoint extension that allows the authors to enrich PowerPoint documents by means of semantic annotation. The key feature was to deal with domain knowledge included in the presentation document as semantic information and to visualize it by means of concept mapping. Ihsan et al. [5] and Verbert et al. [6] proposed e-learning content development tools like PowerPoint/OpenOffice.org. These tools managed not only the domain knowledge but also the information for education/learning using the metadata standards such as IEEE Metadata Standards or SCORM.

Hayama et al. [7] proposed an automatic approach for generating presentation slides from a technical paper. Following a machine learning technique, they could obtain a set of generating rules from the relationships between technical papers and presentation slides collected from the Web. Seta and Ikeda [8] developed a support environment with which the novices can produce persuasive presentation documents and develop their presentation skills. This support environment was designed to encourage the presenter to perform meta-cognitive activities in presentation document design and import expertise of other experienced learners through presentation rehearsal. Li and Chang [9] developed the management model and tools that enable users to better exploit and transfer presentational knowledge assets for representing the domain knowledge.

In spite of the significance of presentation structure, each of these researches and technologies does not deal well with the structure information embedded in the presentation documents. In this paper, we address the issue of how to help novice researchers develop skills in composing the presentation structures.

3 Web Services with Presentation Structure Analysis

3.1 Machine Learning for Presentation Structure Analysis

In order to analyze the presentation structure, we first use the presentation documents accumulated in the laboratory. These documents are brushed up through presentation

rehearsals conducted in the laboratory. The presentation slides often include typical keywords that allow clues for identifying the metadata of the slides in the laboratory. It is accordingly possible to obtain the slide metadata from the typical keywords in the slides to analyze the presentation structure. Considering the typical keywords, we use the machine learning technique to identify the relationships between the slide metadata and typical keywords included in the slides from the presentation documents as training data that are attached in advance with the metadata. Such relationships identified can be used to detect the presentation structure of the documents composed by the laboratory members, which represents the presentation heuristics to be shared in the laboratory.

In the machine learning process, a keyword vector representing each slide metadata is calculated from the presentation documents with the metadata. Detail steps for the keyword vector calculation are as follows:

Step 1. Noun words are extracted by using MeCab (Japanese language morphological analyzer) [10] from each slide in the document attached with the metadata in advance.

Step 2. The keyword vector of the slide metadata is shown by the following formula.

$$V_i = (w_{i,1}, w_{i,2}, \dots, w_{i,j}, \dots, w_{i,m}) \quad (1 \leq j \leq m, 1 \leq i \leq l). \quad (1)$$

where $w_{i,j}$ is the weight score of the word j in the slide metadata i . Each $w_{i,j}$ is calculated by the following expanded *tf-idf* formula.

$$w_{i,j} = tf_{i,j} \cdot imf_j \quad \left(imf_j = \log_{10} \frac{l}{mf_j} \right). \quad (2)$$

where $tf_{i,j}$ is summation of the weight score for the word j included in the slide metadata i of all the documents, which represents appearance frequency of the word j in the slide metadata i . imf_j is inverse metadata frequency of the word j . l is total number of the slide metadata, and mf_j is number of the slide metadata including the word j . imf_j decreases importance of the word j appeared in diverse slide metadata as a general word filter. Finally, each weight score in the keyword vector is normalized by each slide metadata.

Step 3. The machine learning technique calculates to what extent each metadata appears at the normalized position in the sequence of the metadata from the training data. Fig. 2 shows an example of calculations for the metadata appearance. In this example, a target presentation document consists of 19 slides. The normalized appearance position of each slide is determined by dividing the end of the previous slide position by the number of the slides. So, the normalized appearance position of the target slide 3 is 0.11 (=2/19). The metadata appearance ratios at the position 0.11 are calculated by counting metadata of the same position in all presentation documents. In this case shown in Fig. 2, among the four presentation documents, two "Table of Contents" slides and two "Overview" slides appear at the normalized position 0.11. The metadata appearance ratios of "Table of Contents" and "Overview" are consequently calculated as 50% (2/4) respectively.

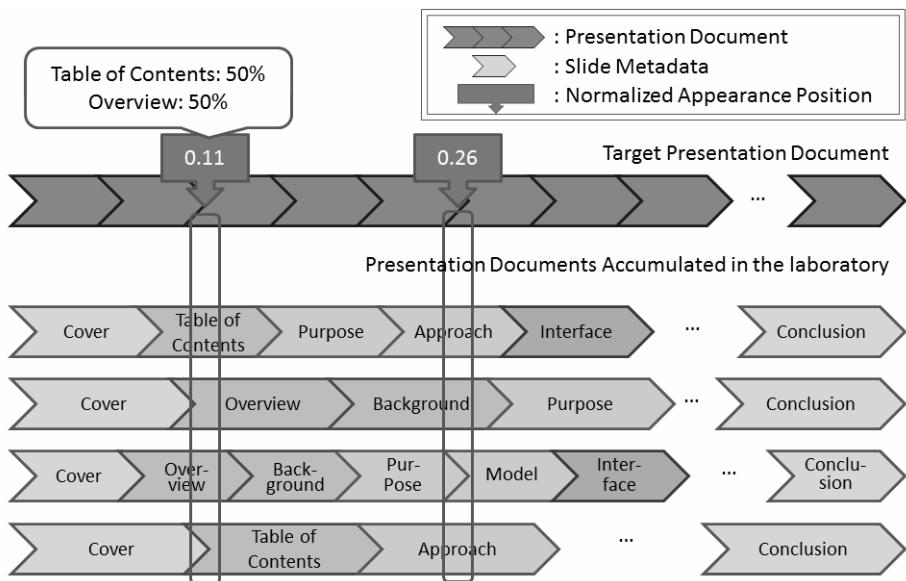


Fig. 2. Normalized Appearance Position

Step 4. This technique also counts the number of the slides included in each segment metadata and calculates averages and standard deviations of the allocation rate of every segment metadata from the documents that have the similar condition in regard to the presentation time. Such allocation rate indicates the presentation schema, which could allow the laboratory members to gain an awareness of presentation structure.

3.2 Recommendation Service for Slide Metadata

Using the results of the presentation structure analysis described above, this Web service recommends the slide metadata to be attached to presentation documents. The aim of this service is to help the novices attach the slide metadata to the documents they compose. The first step towards identifying appropriate slide metadata from the typical keywords included in the slide is to calculate the keyword vector of each slide in the same way as the machine learning technique does. The second step is to calculate degree of similarity between the target slide k and the slide metadata i by means of the following formula as inner product of each keyword vector.

$$Sim(k, i) = V_k \cdot V_i. \quad (3)$$

where V_k and V_i are the keyword vectors of the slide k and the slide metadata i . V_k is calculated in the presentation structure analysis. The candidates of the slide metadata are ranked in a descending order of the similarity degree. The next step is to calculate the normalized appearance position of each slide in the same way as the training data. The candidates of the slide metadata are ranked in a descending order of the normalized appearance frequency. Following these orders, this service extracts metadata candidates and sorts them by multiplying the keyword vector similarity and

normalized appearance frequency, which are recommended as appropriate metadata of the target slide.

3.3 Diagnosis Service for Presentation Structure and Keywords

This service uses the results of the presentation structure analysis to diagnose the presentation structure and keywords embedded in the presentation documents composed by the laboratory members because the results could reflect on the typical presentation structure that represents presentation heuristics shared in the laboratory.

It provides four functions as follows: (1) Segment sequence checker detects fragmentations of the segments estimated from the slide metadata so that the novices can notice discontinuities of the presentation sequence easily. (2) Segment balance checker detects an allocation tendency of the segment metadata by comparing the target document to average and standard deviation of the allocation rate of the training data. This function enables them to adjust the segment allocations in order to compose the presentation documents depending on the presentation time. (3) Metadata keyword checker evaluates whether typical keywords corresponding to certain slide metadata are used in each slide. The typical keywords are determined by the weight score of the word $w_{i,j}$ more than the average word weight score of the slide metadata. (4) Title keyword checker evaluates whether keywords including the title of the presentation document are used in each segment. The presentation title would include most important keywords for the presentation, but the keywords are not included in the slides of the document since it is often forgotten in the process of composing the document. This function accordingly allows them to reflect on appropriateness of the presentation title and slide contents.

3.4 Web Services Interface

These services have been implemented with Microsoft Visual C# 2008, ASP.NET 3.5, SQL Server 2008 and Silverlight 3 in order to run like desktop applications on the major web browsers such as Microsoft Internet Explorer, Mozilla FireFox, and Google Chrome. These services support Microsoft PowerPoint 2007 format (.pptx) as the presentation documents.

The service for the metadata recommendation estimates the slide metadata corresponding to the target slide as shown in left side of Fig. 3. After uploading a .pptx file as the presentation document, a laboratory member can attach the slide metadata to each slide included in the document by means of the recommendation function. When he/she pushes the button for metadata recommendation, Metadata Editor shows the results of the recommendation at the right side of the slide thumbnail. The metadata selected are stored with the document to the Web server.

The service for the structure diagnosis provides him/her with four types of interactive diagnosis functions. Right side of Fig. 3 shows the segment balance checker, which displays pie charts for the allocation rate of the segment metadata. He/she could compare his/her segment allocation data to the average segment allocation data calculated from the presentation documents with the similar condition in the presentation time. These environments enable the novices to improve the documents from a semantics structure point of view.

4 Case Studies

This section describes case studies whose purpose was to investigate whether the services enabled suitable metadata recommendation and structure diagnosis in indirect manners. In these studies, we used 12 presentation documents for the interim presentation (presentation time: 7 minutes) of our laboratory members as training or target data. The main domain of these documents was to develop self-directed learning support systems. The slide metadata were attached to all the documents by a knowledgeable researcher (one of the authors) as correct metadata in advance.

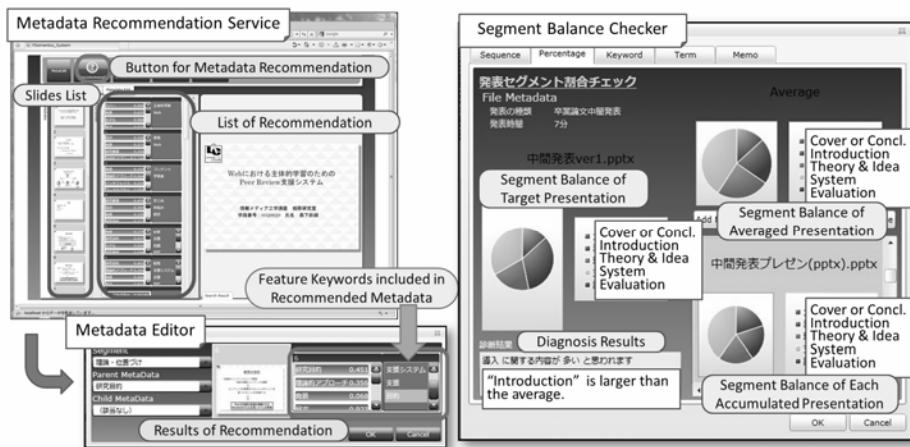


Fig. 3. System Interface

4.1 Case Study for Recommendation Service

The purpose of this study was to compare the accuracy among three recommendation approaches using (1) the keyword similarity, (2) the normalized appearance position, and (3) these combination. First, we chose a certain presentation document from the 12 documents, which deleted the slide metadata attached as a recommendation target. Next, we registered the other 11 documents on the service as training data for the presentation structure analysis and recommended the slide metadata for the target document from the results of the analysis. We repeated such recommendation process for every document and compared the slide metadata attached by the knowledgeable researcher to the ones attached by the service.

Table 1 shows the results of the accuracy of the metadata recommended which were divided into two types, (a) accuracy of first place, which represented the rate of the metadata correctly recommended in the first place, (b) accuracy of top three places, which represented the rate of the metadata correctly recommended in the top three places. Table 1 indicates that the results of the combination approach (3) were more suitable for the metadata recommendation than (1) and (2). From the results of type (a), fully-automated metadata recommendation has still a lot of problems. From the results of type (b), on the other hand, metadata candidates would work to some

extent. Of course, it would be a feature work whether the novices could attach the suitable slide metadata by using the metadata candidates.

A number of reasons could be given for reviewing the failures of the recommendation in case of the recommendation approach (3) as follows: The slide using minor keywords in the research group reduced the accuracy of the recommendation. Some slide metadata such as "Architecture" and "Interface" did not have enough typical keywords since most of them consisted of a couple of pictures or figures. There were some keywords, which had close relationships with several slide metadata. For example, the keyword "task" was associated with not only "Approach (i.e. target tasks)" but also "Conclusion (i.e. future tasks)". Metadata of "Table of Contents" were often used as breakpoints of the presentation document.

Table 1. Accuracy Rate of Metadata Recommendation

	(1)	(2)	(3)
(a)	40.6% (78/192)	42.7% (82/192)	53.1% (102/192)
(b)	62.0% (119/192)	74.0% (142/192)	77.1% (148/192)

4.2 Case Study for Recommendation Service

The purpose of this study was to observe the variation of the presentation structure in refinement processes by means of the proposed diagnosis functions. In this study, we first registered the above 12 presentation documents with metadata on the service as the training data. Then we prepared paired documents different from the training data, which a novice in our laboratory composed. These documents were the first and final versions for the interim presentation. Assuming that the final version was more refined than the first version, we investigated differences in each diagnosis result.

Table 2 shows the diagnosis results of the first and final versions of the document. Comparing these results, problems of the segment sequence, segment balance, and title keyword in the first version were resolved through the presentation refinements. On the other hand, the diagnosis results of metadata keyword showed that they could not use typical keywords for the slide metadata in the final version less than in the first version. These results suggest availabilities of such diagnosis services except for the metadata keyword diagnosis by means of the presentation structure awareness.

Table 2. Diagnosis Results of First and Final Versions of Presentation

Functions	First Version	Final Version
Segment Sequence	1 Fragmentation	0 Fragmentation
Segment Balance	"Introduction" is larger than the average.	All segments are average proportion.
Metadata Keyword	No Typical Keyword Rate $7/15 = 46.7\%$	No Typical Keyword Rate $9/18 = 50.0\%$
Title Keyword	Title Keyword Rate $2/4 = 50.0\%$	Title Keyword Rate $4/4 = 100.0\%$

5 Conclusion

This paper has described the presentation improvement support Web services with the presentation structure framework. The fundamental function of the services is to analyze the presentation structure automatically by the machine learning technique. We have also demonstrated the recommendation service that recommends the metadata to be attached, and the diagnosis service that diagnoses the presentation structure from the documents repository. These services utilize the results of the presentation structure analysis with the machine learning technique.

In addition, this paper has discussed case studies with the proposed recommendation and diagnosis services. The results indicate that these services would make it possible to provide the novices with awareness that could contribute to developing skills in composing presentation documents with presentation heuristics.

In the near future, it will be necessary to improve the recommendation accuracy and to facilitate skill development for composing presentation documents. Furthermore, we will evaluate effectiveness of the diagnosis service for the presentation refinement process in a more detail.

Acknowledgments. This work is supported in part by Grant-in-Aid for Scientific Research (B) (No. 20300266) from the Ministry of Education, Science, and Culture of Japan.

References

1. Hasegawa, S., Tanida, A., Kashihara, A.: Recommendation and Diagnosis Services for Presentation Semantics. In: Proc. of The 18th International Conference on Computers in Education (ICCE 2010), pp. 285–289 (2010)
2. Saito, K., Tanida, A., Kashihara, A., Hasegawa, S.: An Interactive Learning Environment for Developing Presentation Skill with Presentation Schema. In: Proc. of E-Learn 2010, pp. 2696–2703 (2010)
3. Tanida, A., Hasegawa, S., Kashihara, A.: Web 2.0 Services for Presentation Planning and Presentation Reflection. In: Proc. of The 16th International Conference on Computers in Education (ICCE 2008), pp. 565–572 (2008)
4. Kohlhase: Semantic PowerPoint: Content and Semantic Technology for Educational Added-Value Services in MS PowerPoint. In: Proc. of World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA 2007), pp. 3576–3583 (2007)
5. Ihsan, I., Rehman, M., Ahmed, M.U., Qadir, M.A.: S-Point A Semantic Based E-learning Content Development Tool. Journal of Applied Sciences 8(1), 127–133 (2008)
6. Verbert, K., Jovanovic, J., Grasevic, D., Duval, E., Meire, M.: Towards a Global Component Architecture for Learning Objects: A Slide Presentation Framework. In: Proc. of World Conference on Educational Multimedia, Hypermedia and Telecommunications (ED-MEDIA 2005), pp. 1429–1436 (2005)
7. Hayama, T., Nanba, H., Kunifugi, S.: Alignment between a technical paper and presentation sheets using a hidden Markov model. In: Proc. Active Media Technology 2005, pp. 102–106 (2005)

8. Seta, K., Ikeda, M.: Design of an Environment for Developing Presentation Skills. In: Proc. of International Conference on Computer and Education (ICCE 2006), pp. 29–36 (2006)
9. Li, S.T., Chang, W.C.: Exploiting and transferring presentational knowledge assets in R&D organizations. *Expert Systems with Applications* 36, 766–777 (2009)
10. Kudo, T., Yamamoto, K., Matsumoto, Y.: Applying Conditional Random Fields to Japanese Morphological Analysis. In: Proc. of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004), pp. 230–237 (2004)

Topic-Based Recommendations for Enterprise 2.0 Resource Sharing Platforms

Rafael Schirru^{1,2}, Stephan Baumann¹,
Martin Memmel^{1,2}, and Andreas Dengel^{1,2}

¹ German Research Center for Artificial Intelligence,
Trippstadter Straße 122, 67663 Kaiserslautern, Germany

² University of Kaiserslautern,

Gottlieb-Daimler-Straße 47, 67663 Kaiserslautern, Germany

{Rafael.Schirru,Stephan.Baumann,Martin.Memmel,Andreas.Dengel}@dfki.de

Abstract. Companies increasingly often deploy social media technologies to foster the knowledge transfer between employees. As the amount of resources in such systems is usually large there is a need for recommender systems that provide personalized information access. Traditional recommender systems suffer from sparsity issues in such environments and do not take the users' different topics of interest into account. We propose a topic-based recommender system tackling these issues. Our approach applies algorithms from the domain of topic detection and tracking on the metadata profiles of the users' preferred resources to identify their interest topics. Every topic is represented as a weighted term vector that can be used to retrieve unknown, relevant resources matching the users' topics of interest. An evaluation of the approach has shown that our method retrieves on-topic resources with a high precision.

Keywords: Knowledge Management, Enterprise 2.0, Recommender System.

1 Introduction

Nowadays, social media technologies are increasingly often deployed to foster the knowledge transfer in the Enterprise. McAfee introduced the concept of the Enterprise 2.0 as a collection of Web 2.0 technologies for generating, sharing, and refining information [7]. As the amount of content in these information systems grows, there is an increasing need for recommender systems that keep the users informed about resources matching their needs and preferences. However, traditional recommender systems based on collaborative filtering suffer from sparsity issues, particularly in scenarios where the amount of items is much larger than the amount of users. Content-based recommender systems on the other hand tend to recommend similar items only (overspecialization problem) thus not considering a user's full range of interests (e.g., [1]).

We propose an approach providing personalized resource recommendations in Enterprise 2.0 platforms. By applying algorithms from the domain of topic

detection and tracking we identify a knowledge worker's different topics of interest. The method analyzes the metadata profiles of each user's preferred resources and derives per topic a weighted term vector as a label. When the user requests recommendations these vectors are used to query an index in order to find previously unknown resources matching the respective interest topics. The evaluation of our method has shown that the proposed approach can generate topic-based recommendations with a high precision.

The remainder of this article is structured as follows: In Section 2 the ALOE system is described, for which the proposed algorithm has been developed. Section 3 presents the topic-based recommendation method in detail. An evaluation of our approach is presented in Section 4. Related work in the field of topic-based recommender systems is presented in Section 5. Finally, in Section 6 we conclude our findings and present ideas for future work.

2 The ALOE System

ALOE¹ is an Enterprise 2.0 resource sharing platform designed for documents of arbitrary format and their associated metadata [8]. It has been developed in the Knowledge Management Group of the German Research Center for Artificial Intelligence.²

2.1 Functionalities and Metadata

The ALOE system supports sharing of bookmarks and all kinds of files (images, audio, video, office documents, etc.). It provides tagging, commenting, and rating functionalities. Search facilities are offered that provide ranking options taking the usage of resources into account (such as most viewed, highest rated, most commented). A group concept has been implemented that enables users to contact and exchange resources with other users that share similar topics of interest. For the personal organization of resources collections are offered that allow users to group thematically related resources together.

The ALOE system enables the knowledge worker to share content according to her topics of interest. In order to make resources easily retrievable they are annotated with metadata by the community of users. Whenever users add resources to their portfolio they have to annotate a title and tags. Optionally a description, author, and licensing information may also be added.

2.2 Eliciting User Preferences

Recommender systems need to learn about the users' preferences in order to provide them with useful recommendations. There are two possibilities to capture these preferences: First, users can be asked to rate items explicitly on a predefined scale. Second, user preferences can be inferred implicitly by observing the user's

¹ <http://aloe-project.de/AloeView/>

² <http://www.dFKI.de/>

interaction with the system [9]. As explicit ratings impose a cognitive cost, often users are reluctant to vote. For that purpose many systems try to infer rating values implicitly by observing the users. For our recommender system based on the ALOE platform, we consider resource contributions, adding a resource to one's portfolio, and watching the detailed metadata of a resource as implicit positive ratings. Explicit ratings are also considered by our recommender system and are taken over as provided by the users. All ratings are on a five point rating scale with five as the best and one as the worst rating that can be given. Further every action that is associated with a preference expression has a priority value assigned (low, middle, high). In case that several such actions from one user are associated with a resource, the rating value of the action with the highest priority is used. The rating and priority values of each preference relevant user action are as follows:

- View detailed metadata: rating 3, priority low
- Add resource to portfolio: rating 4, priority middle
- Contribute Resource: rating 4, priority middle
- Rate resource: the user's rating value, priority high

3 Topic-Based Resource Recommendations

The goal of our approach is the provision of resource recommendations according to a knowledge worker's different topics of interest. To achieve this target we need to gather a critical amount of information about the users first. Currently, we require that a user has explicitly or implicitly expressed a preference for at least 20 resources. However we want to generate useful recommendations also for users that are new to the system. For that purpose we propose a *switching hybrid recommender system* that generates traditional item-based collaborative filtering recommendations [10] for users for which no interest topics have been identified yet. For users that have interacted with the system over a longer period of time and expressed preferences for a sufficient amount of items we determine the user's topics of interest and use them for content-based recommendations. The current Section describes the calculations that are performed offline as well as the online recommendation generation process.

3.1 Offline Analysis

In order to provide recommendations in real time for many users some time-consuming calculations are performed offline, stored in a data base or an index, and can be retrieved quickly when recommendations are requested. The calculations performed offline will be described subsequently.

Item Similarities. Once every day, we calculate the similarities between all public items in the ALOE data base, for which at least three common users have provided explicit or implicit ratings. The similarity between items is calculated according to the traditional item-based collaborative filtering algorithm as proposed in by Sarwar et al. [10].

User Interests Identification. We identify the knowledge workers' topics of interest by applying textual data mining techniques on the metadata profiles of her preferred resources. The different steps of our topic extraction algorithm will be summarized in short subsequently. A detailed description was previously provided in [11].

Data Access: We determine all resources for which a user has expressed a preference since the last time her interests have been extracted. For each of these resources a metadata profile consisting of the annotated titles and the tags is composed. Per resource potentially many titles are available as every user that adds the resource to her portfolio has to provide a title.

Preprocessing: We convert the terms contained in the metadata profiles to lower case characters, remove punctuation characters and stop words. Further stemming is applied to bring the terms to a normalized form. The normalized profiles of the resources are mapped to a vector space where the features correspond to the terms in the corpus (i.e., the currently considered set of the user's preferred resources) and the feature values are the counts of the words in the respective metadata profiles.

Noise Reduction: Very rare and very frequent terms are not considered helpful to characterize resources. As a consequence dimensions representing these terms are removed.

Term Weighting: Terms that appear frequently in the metadata profile of one resource but rarely in the whole corpus are likely to be good discriminators and should therefore obtain a higher weight. We use the TF-IDF [5] measure to achieve this goal.

Clustering and Cluster Labeling: To be able to cluster the set of a user's preferred resources we need to find a reasonable number of clusters in our data first. For this purpose we follow an approach which is based on the residual sum of squares (RSS) in a clustering result (see [6], page 365). For document clustering and cluster label extraction we apply non-negative matrix factorization (NMF, [13]) on the term-document matrix representing a user's preferred resources. NMF is a co-clustering technique that clusters the rows and columns of a matrix simultaneously. The output of the NMF algorithm is for each term and document its degree of affiliation to each identified topic cluster. By choosing for each topic the ten most relevant terms we obtain for every user interest a weighted term vector characterizing the respective topic. An example of a user profile consisting of many such topics is depicted in Figure 1.

Item Profiles Index. In order to enable a fast lookup of items matching a user's topics of interest we store the metadata profiles of all public resources of the ALOE system in a Lucene³ index. The profiles consist of the titles, descriptions, and tags that have been annotated for each resource. A new index is generated once every day.

³ <http://lucene.apache.org/>

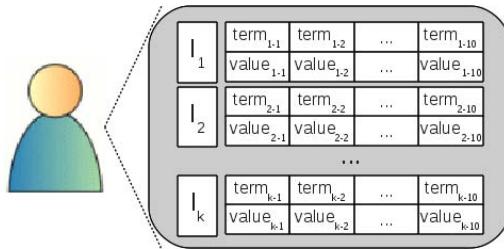


Fig. 1. Profile representing a user's different topics of interest by weighted term vectors \boxed{II}

3.2 Online Recommendation Generation

Whenever a user requests recommendations, two use cases have to be distinguished that will be described subsequently:

Recommendations w/o Interest Topics Available. The user has expressed a preference for at least one resource for which similar items could be determined, however no user interest topics are available yet. This may be accounted to the following reasons: First, it might be that the user has not expressed preferences for the minimum number of required items. Second, it might be the case that the user interests identification algorithm has not been run for the current user yet. In this case recommendations are calculated according to the item-based collaborative filtering method.

Recommendations with Interest Topics Available. In the case that a user's topics of interests have been identified, the recommendations are generated as follows:

- The user's interest topics are loaded from the data base.
- From the most current interest topics we select randomly a predefined number. Currently, we select at most five interest topics. However if less topics are available, all of them will be used.
- The index containing the item profiles is queried according to these interest topics. For each topic, we compose a query that contains the relevant topic terms. A discussion on how to choose the term relevance threshold is provided in Section 4.3. The term weights are used as boosting factors in our query and the terms are connected by an “OR” semantic.
- From the retrieved resources those items which are already known by the user are removed.
- A recommendation list is composed that should contain at most ten items and we aim at delivering an equal number of items for each selected user interest topic.
- Metadata of the recommended items is loaded from the ALOE data base and the list is returned to the user.

4 Evaluation

The purpose of our evaluation study was to examine whether the derived cluster labels could separate a user's resources that are associated with a topic from the rest of the user's resources. A similar approach to evaluate cluster labels for topic-based recommendations has been applied by Yeung et al. [14]. Section 4.1 presents the data set that we used for our evaluation. The results are presented in Section 4.2, followed by a discussion in Section 4.3.

4.1 Data Set

Currently there are not enough users that regularly use ALOE, thus making an expressive evaluation based on the data in the ALOE system impossible. So we decided to use the BibSonomy data set⁴ which is freely available for research purposes. BibSonomy⁵ is a social sharing platform for bookmarks (URLs) and publication references [4]. The provided data set consists of four tables:

- Table *bibtex* stores the bibtex entries that are associated with a shared publication. It has 566,939 instances.
- Table *bookmark* contains the URLs and metadata of bookmark contributions. The data set comprises 275,410 instances.
- In table *tas* the tag assignments of bibtex entries and bookmarks are stored. There are 2,622,423 tag assignments by 6,569 users for 837,757 resources available. For bookmarks only there are 1,009,010 tags provided by 4,095 users available.
- Further in table *relation* sub and super relations of tags are stored. The table has 11,292 instances.

In ALOE the users can share files and bookmarks. We decided to focus on the BibSonomy bookmarks for our evaluation as we consider them to best represent the data and metadata in ALOE. As described in Section 3.1 we use titles and tags to construct metadata profiles of the resources from which the topics are determined. This metadata is also available for resources in the BibSonomy system.

When deploying a topic-based recommender system the current user interest topics should be extracted on a regular basis, e. g., once every week. We selected users that have contributed between 20 and 500 resources as we consider this amount as representative to be contributed in such a time interval. From 503 users matching this criterion we skipped the first 200 early adopters of the system. From the remaining 303 users our approach could detect topics for 296 of them. We analyzed the contributions of the remaining seven users for which no topics could be extracted. It was found that these users either did not provide the required metadata or they contributed spam. In our preprocessing steps the metadata profiles of these resources were deleted.

⁴ Knowledge and Data Engineering Group, University of Kassel: Benchmark Folksonomy Data from BibSonomy, version of July 1st, 2010

⁵ <http://bibsonomy.org/>

4.2 Results

To evaluate how good the cluster labels can separate on-topic from off-topic resources we took for each user and each interest topic all relevant terms from the cluster label and queried the data base with them. We used an exact match query with an “AND” semantic. The selected terms of the topic labels had to appear in the tags, the title, or the description of a retrieved resource. We compared the retrieved resources to the set of resources the clustering algorithm had found for the respective topic. A term was considered as relevant, when it’s relevance value was at least $r\%$ of the relevance value of the most relevant term in the cluster. We controlled parameter r to see which term relevance threshold would provide the best results. The measures precision, recall, and the F-measure have been used to evaluate our approach. These measures are commonly applied to evaluate systems in the field of information retrieval.

Figure 2 plots the precision, recall, and F-measure values of our experiments for different term relevance thresholds. The best results were achieved when only the most relevant terms were used for the query. With a term relevance threshold r of 100% the average precision per user was 0.85, the average recall was 0.42, and the average F-measure was 0.49. When examining the results in greater detail, we found that the accuracy of the clustering algorithm was less than perfect thus leading to off topic resources in the clusters. Such resources are not intended to be found by our algorithm that way leading to decreased recall values. However with an average precision of 0.85 we are confident that our algorithm can recommend resources matching the users’ topics of interest.

4.3 Discussion

The goal of the evaluation study was to show that the cluster labels derived by our approach are capable of separating resources belonging to a topic from the

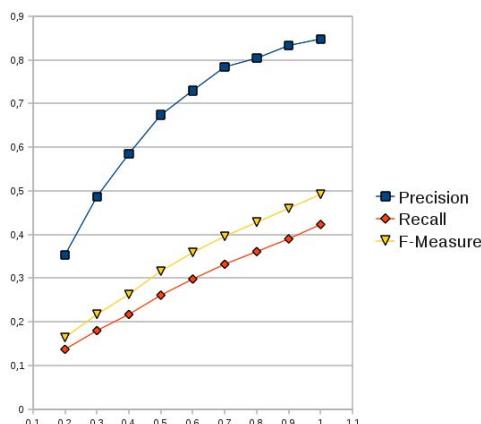


Fig. 2. Evaluation of topic labels by making use of precision, recall, and F-measure, the term relevance threshold has been used as control parameter

rest of the resources. In our experiments we found that by including only the most relevant terms of the label the best results could be achieved. However, a preliminary evaluation study that has been conducted before [11], has shown that users were able to associate cluster labels with their interest topics also when the term relevance threshold was set to 25% only. We therefore propose to relax the condition of using queries with only the most relevant topic terms. Instead, our system uses a query with an “OR” semantic and boosts the terms according to their relevance values that way ranking those items best matching a topic label first. In other scenarios in which our topic extraction approach has been applied, we found that a term relevance threshold of 50% leads to reasonable results [12]. In our future work the overall recommendation quality as perceived by the users has to be tested in a live user experiment.

5 Related Work

A folksonomy-based approach for user interests identification in collaborative tagging systems has been proposed by Yeung et al. [14]. Assuming that the resources and tags posted to such systems highly depend on the users’ interests they use the folksonomies in these systems to build topic-based user profiles. The authors propose a network analysis technique applied on the personomy of the users to identify their different topics of interest. A personomy is defined as the part of the folksonomy that is restricted to the tags, documents, and annotations of one particular user. It is represented as a bipartite graph. This graph is clustered and for each cluster the authors extract a signature consisting of the tags that appear with more than a certain percentage of the documents in the cluster. Finally a user profile is returned as a set of these signatures. A major difference to our approach is that Yeung et al. do not exploit the user-contributed metadata of other users for shared bookmarks. Hence, collective intelligence is not harnessed to obtain proper resource descriptions.

Guo and Yoshi propose a topic-based recommendation framework integrating the tag annotations of individual users, user communities, and all users of a collaborative tagging system [3]. They apply a modified Latent Dirichlet Allocation model [2] to cluster users and tags simultaneously thus obtaining the implicit linking of tags and users. The authors assume a fixed number of 100 topics. They calculate vectors that determine the degree of affiliation of each resource, user, user community, and query term to each topic. Recommendations are then provided by first combining the vectors of the query terms, the active user, and the community of the active user. Second, the top five topics are selected and all bookmarks with a high degree of affiliation for the selected topics are found in the data base. Experimental results show that the proposed recommendation method can alleviate data sparsity and provide more effective recommendations than previous methods. Guo and Joshi assign each user to a topic community based on the user’s interest value for the associated topic. Our method does not assume one predominant interest topic for a user. It retrieves resources for different interest topics each with equal weight.

Ziegler et al. aim at improving topic diversification by balancing top- N recommendation lists according to the users' full ranges of interests [15]. In their recommender system each item is associated with features from a domain taxonomy like, e.g., author, genre, and audience in the domain of books. The proposed algorithm takes a top- N recommendation list and selects a (much) smaller subset of items with a low degree of intra-list similarity. The final recommendation list is built by gradually adding items that keep intra-list similarity low and are recommendable according to traditional collaborative filtering algorithms. The approach presented by Ziegler et al. assumes that features from a domain taxonomy are annotated for each item. In Enterprise 2.0 platforms such features are not always available as many systems do not want to place the burden of annotating content with concepts from a formal taxonomy on the users.

6 Conclusion and Future Work

In this work we have presented a recommender system for Enterprise 2.0 resource sharing platforms taking the knowledge workers different topics of interest into account. By applying a text mining algorithm on the metadata profiles of the users' preferred resources, we derive weighted term vectors that characterize the users' topics of interest. The vectors are used to query an index for items matching these topics. An evaluation of our approach has shown that the proposed method can retrieve resources for a selected topic with a high precision.

In our future work we will have to evaluate how users of the system perceive the quality of the recommendations as well as the diversity of topics. Besides identifying the users' most current topics of interest we are also working on the identification of persistent topics in which knowledge workers are interested for a longer period of time.

Acknowledgments. This research has been financed by the Investitionsbank Berlin in the project "Social Media Miner", and co-financed by the EFRE funds of the European Union.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), 734–749 (2005)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *Journal of Machine Learning Research* 3, 993–1022 (2003)
3. Guo, Y., Joshi, J.B.D.: Topic-based personalized recommendation for collaborative tagging system. In: *HT 2010: Proceedings of the 21st ACM Conference on Hypertext and Hypermedia*, pp. 61–66. ACM, New York (2010)
4. Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: Bibsonomy: A social bookmark and publication sharing system. In: *Proceedings of the Conceptual Structures Tool Interoperability Workshop at the 14th International Conference on Conceptual Structures*, pp. 87–102. Aalborg University Press (2006)

5. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation* 28(1), 11–21 (1972)
6. Manning, C.D., Raghavan, P., Schütze, H.: *Introduction to Information Retrieval*, online edition. Cambridge University Press, Cambridge (April 2009)
7. McAfee, A.P.: Enterprise 2.0: The dawn of emergent collaboration. *MIT Sloan Management Review* 47(3), 21–28 (2006)
8. Memmel, M., Schirru, R.: Sharing digital resources and metadata for open and flexible knowledge management systems. In: Tochtermann, K., Maurer, H. (eds.) *Proceedings of the 7th International Conference on Knowledge Management (I-KNOW)*, Know-Center, Graz, *Journal of Universal Computer Science*, pp. 41–48 (September 2007) ISSN 0948-695x
9. Nichols, D.M.: Implicit rating and filtering. In: *Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering*, pp. 31–36 (1997)
10. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: *WWW 2001: Proceedings of the 10th International Conference on World Wide Web*, pp. 285–295. ACM, New York (2001)
11. Schirru, R., Baumann, S., Memmel, M., Dengel, A.: Extraction of contextualized user interest profiles in social sharing platforms. *Journal of Universal Computer Science* 16(16), 2196–2213 (2010)
12. Schirru, R., Obradović, D., Baumann, S., Wortmann, P.: Domain-specific identification of topics and trends in the blogosphere. In: Perner, P. (ed.) *ICDM 2010. LNCS (LNAI)*, vol. 6171, pp. 490–504. Springer, Heidelberg (2010)
13. Xu, W., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: *SIGIR 2003: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 267–273. ACM, New York (2003)
14. Yeung, C.M.A., Gibbins, N., Shadbolt, N.: A study of user profile generation from folksonomies. In: *Proceedings of the Workshop on Social Web and Knowledge Management, WWW Conf.* (April 2008)
15. Ziegler, C.-N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: *WWW 2005: Proceedings of the 14th International Conference on World Wide Web*, pp. 22–32. ACM, New York (2005)

Custom Ontologies for an Automated Image Annotation System

Gabriel Mihai, Liana Stanescu, Dumitru Dan Burdescu, and Marius Brezovan

University of Craiova, Faculty of Automation, Computers and Electronics,
Bvd. Decebal, No.107, Craiova, Romania
{mihai_gabriel,stanescu,
burdescu,brezovan_marius}@software.ucv.ro

Abstract. Automated annotation of digital images is a challenging task being used for indexing, retrieving, and understanding of large collections of image data. Several machine learning approaches have been proposed to model the existing associations between words and images. Each approach is trying to assign to a test image some meaningful words taking into account a set of feature vectors extracted from that image. In general for the annotation process of medical or natural images the words are retrieved from a controlled vocabulary or from an ontology. This paper presents an original approach for creating two ontologies and an original design of an image annotation system. The ontologies are created using the information provided by two distinct sources: MeSH – a vocabulary used for subject indexing and searching of journal articles in the life sciences and SAIAPR TC-12 Dataset – a set of annotated images having a vocabulary with a hierarchical structure. The annotation system is using an efficient annotation model called Cross Media Relevance Model each image being segmented using a segmentation algorithm based on a hexagonal structure.

Keywords: image annotation, image segmentation, ontology.

1 Introduction

Automatic image annotation is a task that assigns words to images taking into account their semantic content. There are two reasons that are making the image annotation a difficult task: *the semantic gap*, being hard to extract semantically meaningful entities using just low level image features and *the lack of correspondence* between the keywords and image regions in the training data.

There are many annotation models proposed and each model has tried to improve a previous one. These models were splitted in two categories:

- a) Parametric models: Co-occurrence Model [1], Translation Model [2], Correlation Latent Dirichlet Allocation [3]
- b) Non-parametric models: Cross Media Relevance Model (CMRM) [4], Continuous Cross-Media Relevance Model (CRM) [5]

An annotation model annotates an image by providing a set of words that are describing the semantic content of that image. Each word is retrieved from a controlled vocabulary since not all words are properly describing the images from a specific

domain. This constraint is mainly required for images retrieved from medical domain. For example an image containing details about an ulcer should be annotated using specific words that are related to digestive diseases. This requirement can be satisfied by using ontologies.

The term ontology originated as a science within philosophy but evolved over time being used in various domains of computer science. An ontology represents an explicit and formal specification of a conceptualization [19] containing a finite list of relevant terms and the relationships between them. Ontologies are enabling knowledge sharing and support for external reasoning.

For medical domain can be used existing ontologies named Open Biological and Biomedical Ontologies [6] or custom ontologies created based on a source of information from a specific domain. Existing ontologies are provided in formats that are not always easy to interpret and use. A high flexibility is obtained when an ontology is created from scratch using a custom approach.

We have designed an annotation system to handle the annotation task for images retrieved from two distinct sets: a set of images from medical domain and a set of images from nature. Our approach can be extended in a similar way for other sets. The annotation model that was integrated in our system is based on an efficient model called CMRM and it expects to have available annotations at region level. Several studies have shown that this model produces better results than two previous models: the Co-occurrence Model and the Translation Model. Using a set of images from nature [13] or a set of images from medical domain the system learns the joint distribution of the blobs and words. In this context a blob represents a cluster of image regions.

Each new image is segmented using a segmentation algorithm [10] which integrates pixels into a grid-graph. The usage of the hexagonal structure improves the time complexity of the methods used and the quality of the segmentation results.

For the annotation process we needed an ontology for each data set. For this reason we have created two ontologies using an original approach: one ontology was created starting from the information provided by Medical Subject Headings (MeSH) [11] and another one starting from the content of the SAIAPR TC-12 Dataset [13]. SAIAPR TC-12 is a dataset which satisfies the requirements of the CMRM model, to have annotations at region level. There are other public datasets available but these are providing annotations only at image level.

MeSH are produced by the National Library of Medicine (NLM) and contain a high number of subject headings being used for subject indexing and searching of journal articles in MEDLINE/PubMed[12]. Medical terminologies like MeSH are good starting points for semantic description providing the user with a static knowledge reference.

The remainder of the paper is organized as follows: related work is discussed in Section 2, Section 3 contains a description of the annotation model and of the data sets used by the system, Section 4 provides a description of the modules included in system's architecture together with some experimental results and Section 5 concludes the paper.

2 Related Work

Object recognition and image annotation are very challenging tasks. For this reason a number of models using a discrete image vocabulary have been proposed for the image annotation task.

Mori et al. [1] used a Co-occurrence Model in which they looked at the co-occurrence of words with image regions created using a regular grid. The image regions from the training data were clustered into a number of region clusters. For each training image its keywords were propagated to each region. The major drawback of this model is that it assumes that if some keywords are annotated to an image, they are propagated to each region in this image with equal probabilities.

Duygulu et al [2] described images using a vocabulary of blobs. For each image region 33 features such as color, texture, position and shape information were computed. The vector quantized image regions are treated as “visual words” and the relationship between these and the textual keywords can be thought as that between one language, such as French, to another language, such as English.. The regions were clustered using the K-means clustering algorithm into 500 clusters called “blobs”. Similar to the Co-occurrence model, the learned parameters of their model called Translation Model are also the conditional distribution probability table. This model does not propagate the keywords of an image to each region with equal probability. Instead, the association probability of a textual keyword to a visual word is taken as a hidden variable and estimated by the Expectation-Maximization (EM) [20] algorithm. This annotation model was a substantial improvement of the Co-occurrence model.

Jeon et al. [3] viewed the annotation process as analogous to the cross-lingual retrieval problem and used a Cross Media Relevance Model to perform both image annotation and ranked retrieval. This model is finding the training images which are similar to the test image and propagate their annotations to the test image. It is assumed that regions in an image can be described using a small vocabulary of blobs. Blobs are generated from image features using clustering. Based on a training set of images with annotations and using probabilistic models it is possible to predict the probability of generating a word given the blobs in an image. This model can be used to automatically annotate and retrieve images given a word as a query. CMRM is much more efficient in implementation than the above mentioned parametric models because it does not have a training stage to estimate model parameters.

A new generation medical knowledge annotation and acquisition system called SENTIENT-MD (Semantic Annotation and Inference for Medical Knowledge Discovery) is presented in [16]. The system has a semantic annotation and inference platform for precise semantic annotation of medical knowledge in natural language text. Natural language parse trees are semantically annotated and transformed into annotated semantic networks for the purpose of inferring general knowledge from the text. Natural language processing techniques are used to abstract the text into a semantically meaningful representation guided by a domain ontology.

An ontology annotation tree browser (OAT) [7] was created to facilitate the analysis of gene lists. OAT includes multiple gene identifier sets, which are merged internally in the OAT database. For this system were generated novel MeSH annotations by mapping accession numbers to MEDLINE entries. In OAT were harmonized two ontologies: one ontology of medical subject headings (MeSH) and gene ontology

(GO), to enable users to use knowledge both from the literature and the annotation projects in the same tool.

An annotation system called M7MeDe [8] is used for surgical videos. This system creates descriptions in the MPEG-7 [9] standard using MeSH classification based on a mapping between MPEG-7 structural annotation classes onto categories of MeSH descriptors. Each video segment is described either using free text or by attaching keywords from MeSH thesaurus to a selected MPEG-7 structured annotation category like What, What Object , What Action etc.

3 The Annotation Model and Data Sets

The Cross Media Relevance Model is a non-parametric model for image annotation and assigns words to the entire image and not to specific blobs. A test image I is annotated by estimating the joint probability of a keyword w and a set of blobs:

$$P(w, b_1, \dots, b_m) = \sum_{J \in T} P(J)P(w, b_1, \dots, b_m | J). \quad (1)$$

For the annotation process the following assumptions are made:

- a) it is given a collection C of un-annotated images
- b) each image I from C can be represented by a discrete set of blobs

$$I = \{b_1 \dots b_m\}$$
- c) there exists a training collection T , of annotated images, where each image J from T has a dual representation in terms of both words and blobs:

$$J = \{b_1 \dots b_m; w_1 \dots w_n\}$$
- d) $P(J)$ is kept uniform over all images in T
- e) the number of blobs and words in each image (m and n) may be different from image to image.
- f) no underlying one to one correspondence is assumed between the set of blobs and the set of words; it is assumed that the set of blobs is related to the set of words.

$P(w, b_1, \dots, b_m | J)$ represents the joint probability of keyword w and the set of blobs (b_1, \dots, b_m) conditioned on training image J . In CMRM it is assumed that, given image J , the events of observing a particular keyword w and any of the blobs (b_1, \dots, b_m) are mutually independent. This means that $P(b_1, \dots, b_m | J)$ can be written as:

$$P(w, b_1, \dots, b_m | J) = P(w | J) \prod_{i=1}^m P(b_i | J). \quad (2)$$

$$P(w | J) = (1 - \alpha_J) \frac{\#(w, J)}{|J|} + \alpha_J \frac{\#(w, T)}{|T|}. \quad (3)$$

$$P(b | J) = (1 - \beta_J) \frac{\#(b, J)}{|J|} + \beta_J \frac{\#(b, T)}{|T|}. \quad (4)$$

$$P(J) = \frac{1}{|T|}. \quad (5)$$

where:

- a) $P(w|J)$, $P(b|J)$ denote the probabilities of selecting the word w , the blob b from the model of the image J .
- b) $\#(w, J)$ denotes the actual number of times the word w occurs in the caption of image J .
- c) $\#(w, T)$ is the total number of times w occurs in all captions in the training set T .
- d) $\#(b, J)$ reflects the actual number of times some region of the image J is labeled with blob b .
- e) $\#(b, T)$ is the cumulative number of occurrences of blob b in the training set.
- f) $|J|$ stands for the count of all words and blobs occurring in image J .
- g) $|T|$ denotes the total size of the training set.
- h) The prior probabilities $P(J)$ are kept uniform over all images in T being estimated with equation (5). The smoothing parameters α and β were used as: $\alpha = 0.1$ and $\beta = 0.9$.

Medical Subject Headings (MeSH) is a comprehensive controlled vocabulary for the purpose of indexing journal articles and books in the life sciences; it can also serve as a thesaurus that facilitates searching. MeSH's structure contains a high number of subject headings also known as descriptors. The *descriptors* or *subject headings* are arranged in a hierarchy. The tree numbers indicate the places within the MeSH hierarchies, also known as the Tree Structures, in which the descriptor appears. Thus, the numbers are the formal computable representation of the hierarchical relationships. The tree locations carry systematic labels known as *tree numbers*, and one descriptor may have several tree numbers. For example, the descriptor "Digestive System Neoplasms" has the tree numbers C06.301 and C04.588.274. Every descriptor also carries a unique alphanumerical ID called *DescriptorUI* that will not change.

Two important relationship types are defined for MeSH content: hierarchical relationships and associative relationships. Many examples of hierarchical relations are instances of the part/whole and class/subclass relationships. Hierarchical relationships in the MeSH thesaurus are at the level of the descriptor and are seen as parent-child relationships. Associative relationships are used to point out in the thesaurus, the existence of other descriptors, which may be more appropriate for a particular purpose. MeSH content is offered as an xml file named desc2010.xml (2010 version) containing the descriptors and a txt file named mtrees2010.txt containing the hierarchical structure. The hierarchical structure of each category can be established based on the tree number and this observation will be taken into account when establishing the hierarchical relationships between concepts.

SAIAPR TC-12 [13][15] benchmark contains segmented and annotated images that can be used to perform experiments on natural images. It represents an extension of the IAPR TC-12 [14] collection for the evaluation of automatic image annotation methods. Each image was manually segmented using a Matlab tool named Interactive Segmentation and Annotation Tool (ISATOOL). Each region has associated a segmentation mask and a label from a predefined vocabulary of 275 labels. This vocabulary is organized according to a hierarchy of concepts having six main branches: *Humans, Animals, Food, Landscape-Nature, Man-made and Other*.

For each pair of regions the following relationships have been calculated in every image: adjacent, disjoint, beside, X-aligned, above, below and Y-aligned.

The following features have been extracted from each region: area, boundary/area, width and height of the region, average and standard deviation in x and y, convexity, average, standard deviation and skewness.

The dataset contains several folders of images, each folder having the structure presented below:

- a) *images* folder contains the initial images that were manually segmented
- b) *segmentation_masks* folder contains for each image region a file having the extension .mat (Matlab files) that is representing a segmentation mask.
- c) *single_mask_folder* contains a single .mat file per image, representing the mask of the entire image.
- d) *spatial_relationships* contains a file per image with information about the spatial relationships detected between each pair of regions.
- e) *segmented_images_folder* contains manually segmented images.
- f) *features.txt* contains the values of the extracted features from each region. Each record from this file contains a tuple having the following format:
(image index, region index, feature1, feature2, ..., feature28)
- g) *labels.txt* file contains the information needed to identify the words assigned to each image region. Each line in this file contains a tuple having the following format : (image index, region index, word index).
- h) *ontology_path.txt* file contains the path in the ontology for each word associated to a region. Each line contains a tuple with the following format:
(image index, region index, path)

4 System's Architecture and Experimental Results

System's architecture is presented in Figure 1 and contains 7 modules:

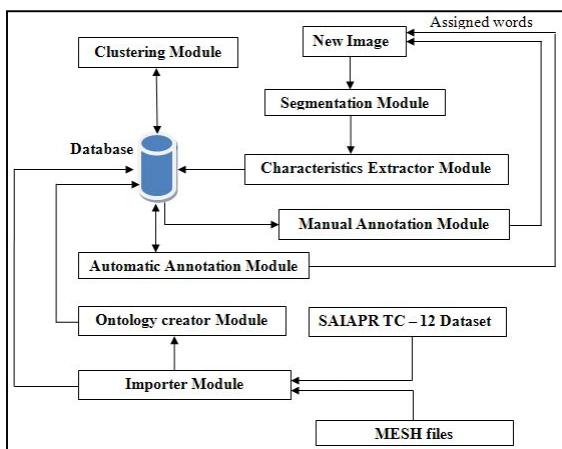


Fig. 1. System's architecture

- a) **Importer module** – this module is used to extract the existing information from the two sources: SAIAPRTC-12 Dataset and MESH files. The import process of MESH content has two steps:
- 1) **Filtering and analyzing the content of desc2010.xml file** – only relevant xml nodes like *DescriptorRecordSet*, *DescriptorRecord*, *DescriptorUI*, *DescriptorName* etc are selected and used.
 - 2) **Analyzing the content of the mtrees2010.txt file** – this file is processed line by line, its content being used to detect the existing hierarchical relationships.

The import process of the SAIAPRTC-12 has three steps:

- 1) **Importing image's regions, features, ontology's paths**
 - 2) **Detecting the spatial relationships between regions** - the content of the *spatial_rels* folder is processed and the spatial relationships are detected.
 - 3) **Importing the list of all words and detecting the words assigned to regions** – the content of the *labels.txt* files is processed and it is detected the word assigned to each region.
- b) **Ontology creator module** - this module can create an ontology for each dataset. The ontology is represented as a Topic Map [17] using the XTM syntax [18]. The mapping process will be described separately for each case:

Descriptor	Topic
<pre><DescriptorRecord> <Descriptor- rUI>D000001</DescriptorUI> <DescriptorName> <String>Calcimycin</String> </DescriptorName> </DescriptorRecord></pre>	<pre><topic id = "D000001"> <instanceOf> <topicRef xlink:href="#concept"/> </instanceOf> <baseName> <baseName- String>Calcimycin</baseNameString> </baseName> </topic></pre>
<pre><DescriptorRecord> <Descriptor- rUI>D001583</DescriptorUI> <DescriptorName> <String>Benzoxazoles</String> </DescriptorName> </DescriptorRecord></pre>	<pre><topic id = "D001583"> <instanceOf><topicRef xlink:href="#concept"/> </instanceOf> <baseName> <baseName- String>Benzoxazoles</baseNameString> </baseName> </topic></pre>
Association	
<pre><association id=" D001583-D000001"> <instanceOf><topicRef xlink:href="#hierarchical"/></instanceOf> <member><roleSpec><topicRef xlink:href="#parent"/></roleSpec> <topicRef xlink:href="#D001583"/></member><member><roleSpec><topicRef xlink:href="#child"/></roleSpec> <topicRef xlink: href="#D000001"/> </member></association></pre>	

- 1) **Mapping for MESH** - in the table 1 it is presented the mapping of two Descriptors (parent with **D001583** as *DescriptorUI* and **D03.438.221** as tree number, child with **D000001** as *DescriptorUI* and **D03.438.221.173** as tree number) to two topic items and the mapping of the hierarchical relationship between them to an association:

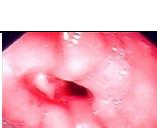
2) **Mapping for SAIAPRTC-12 Dataset**

In the bellow table are presented two ontology concepts (*Mountain* and *Landscape*) modeled as topics and a hierarchical relationship between them modeled as an association:

Word index	Word	Topic
168	Mountain	<topic id = “168”> <instanceOf> <topicRef xlink:href="#concept"/> </instanceOf> <base- Name><baseNameString> Mountain </baseNameString></baseName> </topic>
148	Landscape	<topic id= “148”> <instanceOf> <topicRef xlink:href="#semantic-class"/> </instanceOf> <baseName><baseNameString> Landscape </baseNameString> </baseName> </topic>
Association		
<association id=”148-168”> <instanceOf><topicRef xlink:href="# hierarchical"/></instanceOf> <member> <roleSpec><topicRef xlink:href="#parent"/></roleSpec> <topicRef xlink:href="#148"/> </member><member> <roleSpec><topicRef xlink:href="#child"/></roleSpec><topicRef xlink: href="#168"/> </member></association>		

- c) **Segmentation module** – this module is using the segmentation algorithm described in [10] to obtain a list of regions from each new image.
- d) **Characteristics extractor module** - for each segmented region it is computed a feature vector containing the following items: area, boundary/area, width and height of the region, average and standard deviation in x and y, convexity, average, standard deviation.
- e) **Clustering module** - we have used K-means algorithm to quantize the feature vectors obtained from the training set and to generate blobs.
- f) **Automatic annotation module** – having the set of blobs and for each blob having a list of words we can determine a list of potential words that can be assigned to the image using formulas (3) and (4) of the Cross Media Relevance Model.
- g) **Manual annotation module** – this module can be used to manually annotate images especially from medical domain. We have used this module to annotate a set of training images representing digestive diseases.

In order to evaluate the annotation system we have used a testing set of 400 images that were manually annotated and not included in the training set used for the CMRM model. This set was segmented using the segmentation algorithm mentioned above and a list of words having the joint probability greater than a threshold value was assigned to each image. Then the number of relevant words automatically assigned by the annotation system was compared against the number of relevant words manually assigned by computing a recall value. Using this approach for each image we have obtained a statistic evaluation having the following structure:

Index	Image	RWAA	WMA	Recall
0		sky-blue, sand-beach, ocean	sand-beach, ocean, boat, palm, hut, sky-blue	3/6 = 0.50
1		sky-blue, grass, ocean, cloud	grass, ocean, boat, cloud, sky-blue, branch	4/6 = 0.66
2		sky, mountain, lake	lake, vegetation, mountain, cloud, sky	3/5 = 0.60
3		mountain, sky-blue, sand-dessert	mountain, lake, sand-dessert, sky-blue	3/4 = 0.75
4		esophagitis, esophageal diseases	esophagitis, inflammation, esophageal diseases, gastrointestinal diseases	2/4 = 0.5
5		peptic ulcer, duodenal diseases, intestinal diseases	peptic ulcer, duodenal diseases, intestinal diseases, digestive system diseases	3/4 = 0.75

RWAA represents relevant words automatically assigned and WMA represents words manually assigned. Recall is calculated by dividing the number of words from RWAA to the number of word from WMA.

After computing the recall value for each image it was obtained a medium recall value equal to 0.73

5 Conclusions and Future Work

In this paper we described an original method for creating ontologies that are used by a system to annotate medical and natural images. Both ontologies were created starting from two information sources: MeSH and SAIAPR TC-12 dataset. SAIAPR

TC-12 benchmark contains a large-size image collection comprising diverse and realistic images, including an annotation vocabulary having a hierarchical organization. The CMRM annotation model implemented by the system was proven to be very efficient by several studies. Each new image (medical image, nature image) can be annotated with words taken from the corresponding ontology. Each ontology was represented using the Topic Map standard, each concept being modeled as a topic item and each relationship as an association having a specific type.

Further extensions of the system will include the two models of image retrieval provided by CMRM: Annotation-based Retrieval Model and Direct Retrieval Model.

References

1. Mori, Y., Takahashi, H., Oka, R.: Image-to-word transformation based on dividing and vector quantizing images with words. In: MISRM 1999, First Intl. Workshop on Multimedia Intelligent Storage and Retrieval Management (1999)
2. Duygulu, P., Barnard, K., de Freitas, N., Forsyth, D.: Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2353, pp. 97–112. Springer, Heidelberg (2002)
3. Michael, D.B., Jordan, M.I.: Modeling annotated data. To appear in the Proceedings of the 26th Annual International ACM SIGIR Conference
4. Jeon, J., Lavrenko, V., Manmatha, R.: Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. In: Proceedings of the 26th Intl. ACM SIGIR Conf., pp. 119–126 (2003)
5. Lavrenko, V., Manmatha, R., Jeon, J.: A model for learning the semantics of pictures. In: Proceedings of Advances in Neural Information Processing Systems, NIPS (2004)
6. <http://www.obofoundry.org/>
7. Bresell, A., Servenius, B., Persson, B.: Ontology Annotation Treebrowser: An Interactive Tool Where the Complementarity of Medical Subject Headings and Gene Ontology Improves the Interpretation of Gene Lists. Applied Bioinformatics 5(4), 225–236 (2006)
8. Kononowicz, A.A., Wisniowski, Z.: MPEG-7 as a Metadata Standard for Indexing of Surgery Videos in Medical E-Learning. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 188–197. Springer, Heidelberg (2008)
9. MPEG-7 Overview,
<http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm>
10. Burdescu, D., Brezovan, M., Ganea, E., Stanescu, L.: A New Method for Segmentation of Images Represented in a HSV Color Space. In: Blanc-Talon, J., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2009. LNCS, vol. 5807, pp. 606–617. Springer, Heidelberg (2009)
11. <http://www.nlm.nih.gov/>
12. <http://www.ncbi.nlm.nih.gov/pubmed>
13. Segmented and Annotated IAPR TC-12 dataset,
<http://imageclef.org/SIAPRdata>
14. IAPR TC-12 Benchmark, <http://imageclef.org/photodata>

15. Escalante, H.J., Hernández, C.A., Gonzalez, J.A., López-López, A., Montes, M., Morales, E.F., Enrique Sucar, L., Villaseñor, L., Grubinger, M.: The segmented and annotated IAPR TC-12 benchmark. *Computer Vision and Image Understanding* 114(4), 419–428 (2010)
16. Baoli, L., Ernest, V.G., Ashwin, R.: Semantic Annotation and Inference for Medical Knowledge Discovery. In: NSF Symposium on Next Generation of Data Mining (NGDM 2007), Baltimore, MD (2007)
17. Topic Maps, <http://www.topicmaps.org/>
18. XTM syntax, <http://www.topicmaps.org/xtm/>
19. Gruber, T.R.: A translation approach to portable ontology specifications. *Knowledge Acquisition* 5, 199–220 (1993)
20. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B (Methodological)* 39(1), 1–38 (1977)

MetaProPOS: A Meta-Process Patterns Ontology for Software Development Communities

Nahla Jlaiel and Mohamed Ben Ahmed

RIADI Laboratory, National School of Computer Science,

University of La Manouba, La Manouba 2010, Tunisia

{Nahla.Jlaiel,Mohamed.Benahmed}@riadi.rnu.tn

Abstract. Process patterns are a valuable mechanism to capture and disseminate best practices during any software development process. Consequently, they have been successfully and increasingly used within software development communities to reuse proven solutions. But the multiplicity, diversity and widespread dissemination of their descriptions are making barriers to identify relevant patterns for a specific context. Hence, a more formal and unified representation of process patterns is needed to allow a rigorous reasoning process as well as machine processing means in order to provide architectural and semantic unification of patterns knowledge in addition to an intelligent interconnection of patterns that are most appropriate to solve a given problem. In this context, we propose a semantic representation based on a process pattern meta-model for which an ontology-based approach is adopted. This latter provides a formal and shared conceptualization as well as a robust inferential basis for building an intelligent framework of process patterns capitalization and reuse within software development communities improving therefore, their ability to develop high quality software.

Keywords: Software Process Patterns, Pattern-based Software Development, Ontologies, Meta Process Pattern's Ontology, OWL ontology.

1 Introduction and Motivation

The pattern concept was introduced for the first time by C. Alexander *et al.* [1] in 1977 as a mechanism to capture knowledge about proven solutions to recurring problems in the urban architecture. The well-cited and generic definition of an Alexandrian pattern as “a solution to a problem in a context”, leads to an increased adoption of patterns by many other domains, including Software Development, Human Computer Interaction, Security, Organization, Pedagogy, to name just a few.

Within the software development community, patterns have been increasingly recognized as an effective method to reuse knowledge and best practices gained during the whole software lifecycle [2] [3]. Thus, software patterns now exist for a wide range of topics including process patterns, analysis, design, implementation or code patterns, test patterns and even maintenance patterns.

Concerning process patterns, they are widely used by the software development community as an excellent medium to share software development knowledge that is

often encapsulated in experiences and best practices [4] [5] [6]. In other words, they capitalize good specifications or implementations of a method to be followed to achieve a goal [7].

As consequence to the huge proliferation of the process patterns practice, several description models and languages were proposed and used in software development research and practice [8]. These latter suffer from serious problems related to eight lacks [8] that we observed and classified into four levels (cf. section 2.1). As a matter of fact, process patterns are often being used in an informal manner, through traditional textbooks or better with modest hypertext systems providing weak semantic relationships. In addition to the huge number of process patterns that are available in books or Web-based resources [3], they significantly differ in format, coverage, scope, architecture and terminology used.

All of these observations conspire to create barriers to the efficient practice of process patterns. Indeed, patterns users are expected to investigate different patterns resources such as books, magazines, papers and Web collections to find the most appropriate patterns. This investigation really needs cognitive efforts, abilities and time to identify, understand, select, adapt and apply relevant ones. For these reasons, we argue that more formal methods and tools are needed to help software development communities use, reuse and capitalize process patterns during any given software development lifecycle.

The overall motivation of our research is to build up an intelligent framework in order to ease process patterns capitalization and reuse [8]. A first step in this direction is to represent different process patterns in a unified and formal manner in order to allow rigorous reasoning process on process patterns' knowledge whatever the format and the terminology used are. In this paper, we propose a meta-model for process patterns' description and representation. To achieve this goal, an ontology-based approach is performed providing common and shared architecture, terminology and semantic for better patterns' unification, mediation and mining.

The remainder of this paper is organized as follows: section 2 provides background information on first, process patterns formats' key observations and second on, ontology-based approaches dealing with software patterns. Section 3 gives details on how the proposed Meta Process Patterns' Ontology for Software development named MetaProPOS, is built and emphasizes the importance of the presented ontology by providing an overview of the work in progress. Section 4 concludes the paper by giving a discussion of our contributions and some future directions.

2 Background

In this section, we begin by reporting some results of the theoretical survey that we have carried out on eleven process patterns' formats (models and languages) found in the literature. This should motivate our choice for an ontology-based approach which is described in the second subsection.

2.1 Process Patterns' Formats

Different works have been carried out in the literature of patterns dealing with process patterns' description and formalization. These are classified into description models

such as AMBLER [9], RHODES [10], GNATZ [11], P-SIGMA [7], STÖRRLE [12], to name just a few and other as languages, such as PROMENADE [13], PPDL [14], PROPEL [15], PLMLx [16], UML-PP [6] and PPL [17].

Based on eleven proposed evaluation criteria [8], we assessed the aforementioned works [8] and revealed the following challenges that we classified at four levels, named FRUGAL^{evels}:

Patterns' Formalization (F). A current challenge is to formalize relevant parts of a process pattern description eg., pattern's problem, context, solution and relationships in order to help find and reuse suitable patterns. In reality, most of the process patterns are described in an informal manner which creates barriers to infer possible potential similarities, interactions, couplings and contradictions among different process patterns.

Knowledge Representation (R). Another revealed challenge is to unify the different pattern conceptualizations that we have discovered in pattern's scope, coverage as well as pattern's relationships whether it is a product or process pattern. In other words, we have found that the assessed languages and models potentially differ in their pattern's knowledge focus, content, structure and degree of detail. This may obstruct or even ruin opportunities to investigate patterns' knowledge machine processing abilities.

Patterns' Unification (U). Another important perceived challenge is to be able to ensure architectural as well as semantic mediation between different process patterns' collections. Indeed, these two kinds of mediation would help to create a unified body of knowledge about process patterns. The architectural unification level aims to normalize different process patterns' descriptions used in the literature and the practice by means of a set of normalized terms. These refer to common concepts that are supported by the different surveyed works and are annotated by a set of derived terms matching those found in the survey. The semantic unification aims to extract terms and/or concepts that are most significant for the process pattern description's content. These are weighted and sorted building thus, semantic annotations for the pattern description's content.

Pattern's Guidance and Assistance (GA). This major challenge consists in providing more guidance and assistance for patterns' users. In fact, among the surveyed works, only few of them accorded importance and supported means for patterns' comprehension and reuse by providing some examples, remarks, guidelines, known uses, application constraints, adaptation parameters, to name just a few. In addition, very few of them offer process patterns support method and/or tool to improve patterns' creation, adaptation, classification, search, mining, reuse, or even enhancement.

To summarize, we have noticed some critical dissents that could, in our opinion, inhibit process patterns' knowledge sharing and management. These could be characterized by terminological, architectural, knowledge and semantic dissent. For this reason, we assume that unification and mediation efforts are needed to reach a consensus so as to help improving process patterns capitalization and reuse. For this purpose, we have adopted an ontology-based approach which aims to afford a formal conceptualization and representation as well as a shared semantic of software process

pattern knowledge, thus allowing robust inferential mechanisms to be developed and usefully explored in order to achieve the objective in view.

2.2 Ontologies and Software Patterns

As aforementioned, the objective of our approach is to provide a shared conceptualization of process patterns' knowledge by building architectural, terminological and semantic mediation facilities between different process pattern languages and collections. To achieve this goal, our approach is based on the use of ontologies. In the following, the concept of ontology is briefly reminded followed by some related works.

Ontology. As widely defined in the literature as “an explicit specification of a conceptualization” [18], an ontology provides a common vocabulary of concepts, relationships as well as axiomatic definitions and formal logic to support machine-based reasoning. Nowadays, ontologies are considered as a popular tool to represent common knowledge about a domain and are widely used for knowledge management and to support semantic search. In fact, traditionally engineered systems have long been used to process the structure or syntax of data and could not process its meaning. Nevertheless, the use of ontologies, should make the semantic available in an explicit and declarative manner helping thus, programs and humans to share and reuse knowledge bodies. Very briefly, it is worth reminding ourselves of the key benefits provided by representing meaning of data being processed [19] [20]:

Interoperability. In order to help heterogeneous systems interoperate with each others, ontologies can be used to build up a shared semantic framework which establishes mappings between different concepts in different systems. In the context of patterns, different collections should coexist and be coupled thanks to ontologies' potentials of heterogeneous information resources integration as well as to its knowledge discovery capabilities.

Well-defined Semantics. Ontologies' use should provide precise definitions of concepts allowing us to know precisely the meaning of domain specific terms and concepts. In the context of process patterns, an ontology-based approach could help to resolve ambiguities of interpretation of patterns described in informal and different manners. So, patterns' semantic data would be explicitly available, making thus patterns machine readable and processable.

Browsing/Searching. The meta-knowledge held by an ontology can assist an intelligent search engine while processing a pattern user query. In this direction, ontologies could not only assist for retrieval but also for discovery by automatically generalizing the query to find nearest partial matches or by generating different views to select the appropriate patterns. Indeed, they can be used to deliver significantly improved semantic search and browsing.

Automatic and Rigorous Reasoning. By using ontologies, concepts and their relationships can be represented. So, automated reasoning could be conducted by inference engines and as a result, it will be possible to infer new relationships to arrive at logical

consequences. This is provided by formal axioms and formal logic on which ontology's languages such as OIL, DAML+OIL and OWL are based.

Related Work. Only few works have been carried out dealing with ontologies and patterns in general. Regarding process patterns, no initiatives have been taken to improve process patterns reuse in spite of all the efforts of formalization. These latter do not lead to a uniform process pattern specification language. In the contrary, they are focalized on particular and private collections of process patterns and do not pay attention to any pattern reuse from other collections. Even, apart from process patterns, we have found two ontology-based works carrying out on patterns.

The first one deals with usability design patterns [3] [21] proposing an ontology-based meta-model for the formal representation of usability design patterns languages. The proposed ontology, called PFOWL (Pattern forms in OWL), aims to provide a semantic framework for pattern-based software development tools. However, small steps have been taken in this work and no continued research as the ontology validation nor its operationalization have been found in this direction since 2007.

Regarding the second initiative [22], it adopts an ontology-based approach to propose a semantic annotation tool for hypermedia patterns which are commonly used in hypermedia design such as web applications. This work is more mature than the first one since it provides a semantic framework allowing annotation of textual hypermedia patterns for domain experts or novices. Although, it adopts a single and particular pattern format and needs human efforts and availability to proceed to the annotation process. In this context, we believe that the better is to create a semantic framework supporting automatic processing of different pattern formats, thus helping the automatic extraction of semantic annotations of patterns.

Apart from these attempts, we have observed that more attention was paid to design patterns among all other patterns' types. Indeed, most of the research works dealt with design patterns and were built on formal specifications of object-oriented languages such as LePUS [23] and its extension eLePUS [24], DisCo [25], BPSL [26]. As Heninger *et al.* observed in [3], all of these formal methods are based on object-oriented systems and do not scale to other patterns types such as process or usability patterns. In addition, no further details or information have been provided on how any reasoning process is performed or how the benefits of formally described patterns could be fulfilled. Recently in 2010, a new workshop was held dedicated to Pattern-driven Engineering of Interactive Computing Systems (PEICS) [27] addressing issues related to Human Computer Interaction patterns. Another major observation that we have noticed is that patterns are mostly used as a valuable means for ontologies' research improvement [28]. Nevertheless, ontologies are rarely applied for patterns' use enhancement. Hence, we think that we should investigate in these two key directions to improve research in patterns as well as ontologies' fields.

3 MetaProPOS: A Unified Conceptualization of Software Process Patterns

As we stated previously, the overall goal of our research is to provide a semantic and reasoning framework in order to support and improve process patterns reuse and

capitalization within software development communities regardless of the adopted software development lifecycle. Because of patterns formats diversity, we propose to set up mechanisms to infer a meta process pattern unifying all the process patterns descriptions, thus providing intelligent interconnection and composition of relevant and unified bodies of knowledge. To formalize and operationalize this, we adopted an ontology-based approach ensuring interoperability between different process patterns collections via a shared and well-formed semantic and providing an automatic and rigorous reasoning process on patterns knowledge. In this section, we present the mapping efforts carried out to unify process patterns' knowledge representations. Based on the mapping results, a discussion of the observations made is addressed followed by an overview of our proposed ontology, named MetaProPOS.

3.1 Mappings between Process Patterns Forms

A first step in this direction is to compare knowledge coverage in the different patterns forms that we identified in the state of the art and practice of process patterns.

To show this effort, we established a mappings table highlighting the different terminologies used to express pattern's knowledge observed in eleven process patterns formats. Because of space limitations, Table 1 is used for illustrative purpose presenting an excerpt of the mappings work. In Table 1, columns indicate the chosen patterns formats, namely: GNATZ, STÖRRLE, PROMENADE, PROPEL and PPL. The table rows show the different terms used to describe a given pattern facet if it is supported or not (--). Indeed, when comparing the chosen patterns formats, we identified eleven key facets. The classification facet indicates how a pattern is organized (by category, type, abstraction level, aspect). The identification facet encapsulates a set of properties identifying a pattern such as pattern name, author(s) and keywords. The problem, context and solution facets refer to the core pattern knowledge. The relationships facet expresses how a pattern interacts with other patterns. The roles facet specifies the responsibilities implied in a pattern application. The artifacts facet gives reference to deliverables that are used and/or produced by a pattern. The guidance facet refers to the support level provided by a pattern to be comprehended and used. The evaluation facet gives feedbacks on pattern application. The management facet provides general information about a given pattern. Only six facets are shown through Table 1 lines.

3.2 Observations and Discussion

Through these comparisons, we have been able to estimate the different facets weights for each process pattern format. Table 2 provides an excerpt of the assessment results where the values 1, 0.8, 0.5, and 0 are respectively assigned for a very well supported facet, a well supported facet, a quite well supported facet and a not supported facet. When observing these results, we deduced the coverage percentage of each facet as Fig. 1 shows. As illustrated in Fig. 1, most of the studied process patterns' formats have paid attention to the four main facets: context, solution, problem and relationships (15%, 14% and 13%). Least attention has been paid to the facets of identification, classification, guidance and roles (11%, 8% and 6%). Very little interest was accorded from to the facets of artifacts, evaluation and management (4%, 3% and 2%).

Table 1. An excerpt of the mappings' table

		GNATZ	STÖRRLE	PROMENADE	PROPEL	PPL
Classification	--	Classification: Abstraction level Phase Purpose Scale	--	Catalog Aspect	--	
Problem	Problem Intent	Intent	Intent	Problem Sub-problem	Problem Intent	
Context	Context	Applicability Consequences	Initial context Result context	Initial context Resulting context	Context Consequence	
Solution	Solution	Process Sample execution	Process	Process	Solution Process Activity Rule	
Relationships	See Also	Related patterns Required Similar Consequent Alternative	Related patterns	pat-	Relationship Sequence Use Refinement Process variance	Related pattern: Use Extend

On the other side of the observations, it can be inferred from Fig. 2 that PLMLx, STÖRRLE and then PROPEL have the most appreciable coverage levels of the eleven proposed pattern facets. In addition, Fig. 2 reveals that six process pattern's formats from eleven cover more than 50% of the facets including PPL, PPDL and PROMENADE. In addition to all these observations, we have to notice the terminological disparity being revealed from mappings between the studied process patterns' formats. This latter will be illustrated and handled through the proposed ontology detailed hereafter.

3.3 Overview of the Proposed Ontology: MetaProPOS

The purpose of MetaProPOS is to interconnect different process patterns' collections whatever the pattern's format is or the terminology employed is in order to uniformly express and share patterns knowledge. To build up our ontology, we adopted a common process [29] including three main phases: conceptualization, ontologization and operationalization. In this subsection, we will just focus on the operationalization result which is the formal representation of MetaProPOS. In this phase, we formalized the proposed conceptualization and the meta-model using the formal Ontology Web Language, OWL. In this context, we used the Protégé 2000 as ontology editor and the OWL Description Logics (OWL-DL) as OWL sublanguage [30] in order to formally define process patterns' knowledge. Indeed, OWL-DL is based on decidable fragments of first order logic as well as axiomatic definitions that could be employed by engines to infer new facts and check ontology's consistency. Fig. 3 depicts a

simplified view of the proposed concepts' hierarchy in which concepts as well as terms are represented through OWL classes and subclasses. For example, Fig. 3 reveals that the terms *Intent*, *Intention*, *Sub-problem* and *Problème* are used to refer to the concept of *Problem*.

Table 2. An excerpt of the assessment's table

	AMBLER	RHODES	P-SIGMA	PPDL	PLMLx	UML-PP
Problem	0	0,8	0,8	1	0,8	1
Context	1	0,5	0,5	1	1	1
Solution	1	1	1	0,8	0,8	0,8
Classification	0,8	0	0,5	0,5	1	0,8
Identification	0,5	0,5	0,5	0,5	1	0
Relationships	0,5	0	1	1	1	1
Roles	0	0	0	0	0	0
Artifacts	0	0	0	0	1	0
Guidance	0,5	0,5	1	0,5	1	0
Evaluation	0	0	0	0,5	1	0
Management	0	0	0	0	1	0

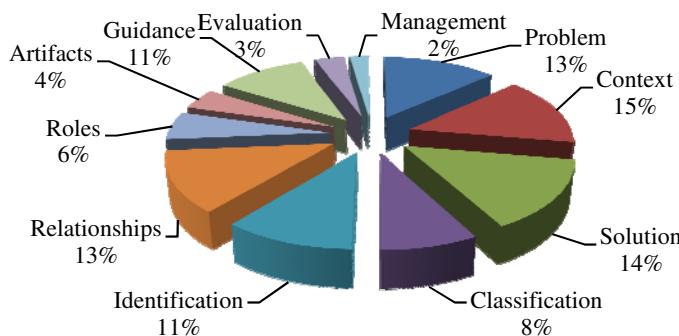


Fig. 1. Facets coverage percentages

Constraints and concepts' properties are handled through the OWL restrictions expressing the mappings rules between the different process patterns formats and the ontology's concepts. Indeed, OWL includes vocabulary for describing classes and properties allowing the construction of class taxonomies. OWL properties act as predicates representing RDF triplets between classes. OWL offers also, axiomatic constructs ensuring a greater expressiveness through quantifiers (existential or someValuesFrom: \exists and universal or allValuesFrom: \forall), qualified cardinalities and general axiomatic definitions of class membership through complex expressions. The Problem of synonymy has been handled in MetaProPOS by means of complex

restrictions such as those dealing with the use of the terms: *Sub-problem*, *Intent* and *Intention* that refer to the pattern Problem's concept. Regarding polysemy, which characterizes terms that are used for different purposes, we added some OWL constraints. These are illustrated by the following example dealing with polysemy caused by the use of the term *Consequences* in PPL to refer to a *Resulting context* as well as *Guidance* in GNATZ. The corresponding OWL fragment code is shown in Fig.4.

$$\forall X \text{ hasFormat}(X, \text{PPL}) \cap \text{pattern}(X) \rightarrow \text{refersTo}(\text{Consequences}, \text{Resulting_context})$$

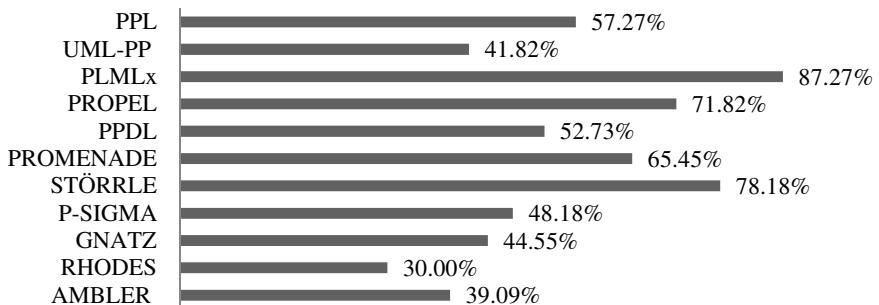
$$\forall X \text{ hasFormat}(X, \text{GNATZ}) \cap \text{pattern}(X) \rightarrow \text{refersTo}(\text{Consequences}, \text{Guidance})$$


Fig. 2. Facets coverage percentages per pattern format



Fig. 3. A fragment of the MetaProPOS concepts hierarchy

```

<owl:Class rdf:ID="#Consequences">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#refersTo"/>
      <owl:allValuesFrom>
        <owl:Class>
          <owl:intersectionOf rdf:parseType="Collection">
            <owl:Restriction>
              <owl:onProperty rdf:resource="#hasFormat"/>
              <owl:hasValue rdf:resource="#PPL"/>
            </owl:Restriction>
            <owl:Restriction>
              <owl:onProperty rdf:resource="#refersTo"/>
              <owl:allValuesFrom rdf:resource="#Resulting_context"/>
            </owl:Restriction>
          </owl:intersectionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#refersTo"/>
      <owl:allValuesFrom>
        <owl:Class>
          <owl:intersectionOf rdf:parseType="Collection">
            <owl:Restriction>
              <owl:onProperty rdf:resource="#hasFormat"/>
              <owl:hasValue rdf:resource="#GNATZ"/>
            </owl:Restriction>
            <owl:Restriction>
              <owl:onProperty rdf:resource="#refersTo"/>
              <owl:allValuesFrom rdf:resource="#Guidance"/>
            </owl:Restriction>
          </owl:intersectionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#Resulting_context"/>
</owl:Class>

```

Fig. 4. An example of the MetaProPOS OWL restrictions

4 Conclusion and Future Work

In this paper, we have shown how a semantic approach could be investigated to improve software process patterns reuse and capitalization within diverse software development communities and for any adopted software development method whether it is agile or not.

The most valuable contribution of this paper is the proposed ontology MetaProPOS which aims to provide a semantic mediation between different process patterns collections. This mediation implies architectural as well as terminological mediation efforts ensured by the proposed ontology. Indeed, the overall goal of MetaProPOS is to help create a *patterns' algebra* allowing rigorous and automatic patterns processing providing better search of similar patterns as well as efficient guidance for patterns composition and aggregation whatever the format and terminology used are. So, a mega process pattern could be created from different relevant parts of process patterns thanks to MetaProPOS.

As a matter of fact, the research work in which MetaProPOS is integrated, aims to create an intelligent framework based on a process patterns warehousing and mining approach to better use and manage best practices and software process implementation traces that are captured and embedded in process patterns [8]. Therefore, MetaProPOS forms the building blocks of our overall approach and continued research is needed to validate our position.

To finish, we have to notice that MetaProPOS is not yet completed and should be iteratively and incrementally built. In this context, we are working on creating a representative corpus of different process patterns in order to experiment, evaluate and enhance MetaProPOS. We should also mention that the proposed solution could be generalized to be adopted and adapted for others software patterns types.

References

1. Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I., Angel, S.: *A Pattern Language: Towns, Buildings, Construction*. Oxford University Press, New York (1977)
2. Buschmann, F., Henney, K., Schmidt, D.C.: *Pattern-oriented Software Architecture: On Patterns and Pattern Languages*. Wiley & Sons, Chichester (2007)
3. Henninger, S., Corrêa, V.: Software Pattern Communities: Current Practices and Challenges. In: ACM Proceedings of 14th International Conference on Pattern Languages of Programming, New York, pp. 1–19 (2007)
4. Hagen, M.: Support for the Definition and Usage of Process Patterns. In: 7th European Conference on Pattern Languages of Programs, Dortmund (2002)
5. Hagen, M., Gruhn, V.: Process Patterns - a Means to Describe Processes in a Flexible Way. In: 5th International Workshop on Software Process Simulation and Modeling, ICSE Workshops, Scotland, pp. 32–39 (2004)
6. Tran, H.N., Coulette, B., Dong, B.T.: Modeling Process Patterns and Their Application. In: IEEE Proceedings of 2nd International Conference on Software Engineering Advances, Cap Esterel, pp. 15–20 (2007)
7. Conte, A., Fredj, M., Giraudin J.P., Rieu, D.: P-Sigma: A Formalism for A Unified Representation of Patterns. In: 19^e Congrès Informatique des Organisations et Systèmes d'Information et de Décision, Martigny, pp. 67–86 (2001) (in French)
8. Jlaiel, N., Ben Ahmed, M.: Reflections on How to Improve Software Process Patterns Capitalization and Reuse. In: 9th International Conference on Information and Knowledge Engineering, pp. 30–35. CSREA Press, Las Vegas (2010)
9. Ambler, S.W.: *Process Patterns: Building Large-Scale Systems Using Object Technology*. Cambridge University Press/SIGS Books, Cambridge (1998)
10. Coulette, B., Crégut, X., Dong, T.B., Tran, D.T.: RHODES, a Process Component Centered Software Engineering Environment. In: 2nd International Conference on Enterprise Information Systems, Stafford, pp. 253–260 (2000)
11. Gnatz, M., Marschall, F., Popp, G., Rausch, A., Schwerin, W.: Towards a Tool Support for a Living Software Development Process. In: Ambriola, V. (ed.) EWSPT 2001. LNCS, vol. 2077, pp. 182–202. Springer, Heidelberg (2001)
12. Störrle, H.: Describing Process Patterns with UML. In: Ambriola, V. (ed.) EWSPT 2001. LNCS, vol. 2077, pp. 173–181. Springer, Heidelberg (2001)
13. Ribó, J.M., Franch, X.: Supporting Process Reuse in PROMENADE. Research report, Politecnical University of Catalonia (2002)
14. Dittmann, T., Gruhn, V., Hagen, M.: Improved Support for the Description and Usage of Process Patterns. In: 1st Workshop on Process Patterns, 17th ACM Conference on Object-Oriented Programming, Systems, Languages and Applications, Seattle, pp. 37–48 (2002)
15. Hagen, M., Gruhn, V.: Towards Flexible Software Processes by using Process Patterns. In: 3rd IASTED Conference on Software Engineering and Applications, Cambridge, pp. 436–441 (2004)

16. PLMLx,
http://www.cs.kent.ac.uk/people/staff/saf/patterns/diethelm/plmlx_doc
17. Meng, X.X., Wang, Y.S., Shi, L., Wang, F.J.: A Process Pattern Language for Agile Methods. In: 14th Asia-Pacific Software Engineering Conference, Nagoya, pp. 374–381 (2007)
18. Gruber, T.: Toward Principles for the Design of Ontologies used for Knowledge Sharing. *International Journal of Human-Computer Studies* 43, 907–928 (1995)
19. Menzies, T.: Cost Benefits of Ontologies. *Intelligence Magazine* 10, 26–32 (1999)
20. Bürger, T., Simperl, E.: Measuring the Benefits of Ontologies. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM-WS 2008. LNCS, vol. 5333, pp. 584–594. Springer, Heidelberg (2008)
21. Henninger, S., Ashokkumar, P.: An Ontology-Based Metamodel for Software Patterns. In: 18th International Conference on Software Engineering and Knowledge Engineering, San Francisco, pp. 327–330 (2006)
22. Montero, S., Díaz, P., Aedo, I.: A Semantic Representation for Domain-Specific Patterns. In: Kock Wiil, U. (ed.) MIS 2004. LNCS, vol. 3511, pp. 129–140. Springer, Heidelberg (2005)
23. Eden, A.H., Hirshfeld, Y.: Principles in Formal Specification of Object-Oriented Design and Architecture. In: 4th Conference of the Centre for Advanced Studies on Collaborative Research, Toronto (2001)
24. Raje, S., Chinnasamy, S.: eLePUS-A Language for Specification of Software Design Patterns. In: 16th ACM Symposium on Applied Computing, Las Vegas, pp. 600–604 (2001)
25. Mikkonen, T.: Formalizing Design Patterns. In: 20th International Conference on Software Engineering, Kyoto, pp. 115–124 (1998)
26. Taibi, T., Ling Ngo, D.C.: Formal Specification of Design Patterns - A Balanced Approach. *Journal of Object Technology* 2(4), 127–140 (2003)
27. Pattern-Driven Engineering of Interactive Computing Systems (PEICS 2011),
<http://www.zmmi.de/PEICS2011/index.html>
28. Ontology Design Patterns (ODPs), http://ontologydesignpatterns.org/wiki/Main_Page
29. Kassel, G., Perrette, S.: Co-operative Ontology Construction Needs to Carefully Articulate Terms, Notions and Objects. In: Proceedings of the International Workshop on Ontological Engineering on the Global Information Infrastructure, Dagstuhl Castle, pp. 57–70 (1999)
30. OWL Web Ontology Language Overview,
<http://www.w3.org/TR/2004/REC-owl-features-20040210/>

An Ontology-Based Framework for Collaborative Maintenance Planning

Ren Genquan^{1,3}, Zhang Yinwen², Zhang Li², Wang Jianmin², and Lan Ting²

¹ Dep. of Computer Science & Technology, Tsinghua University, Beijing, China

² School of Software, Tsinghua University, Beijing, China

³ Logistical Scientific Research Institute, Beijing, China

rgq07@mails.tsinghua.edu.cn, lizhang@tsinghua.edu.cn

Abstract. Three essential elements for formulating the maintenance planning are the choice of maintenance policy, defining maintenance procedure and allocation of maintenance resources. The granularity of computer-aided maintenance planning mainly depends on the representation granularity of maintenance procedure. After analyzing the typical mode to make computer-aided maintenance planning, we put forward one ontology-based framework for collaborative maintenance planning, whose concepts are divided into maintenance knowledge category and maintenance business category. Then the concepts and the relationships among them are explored through the ontology models in concept layer and one example in instance layer. Finally some models are illustrated using ontology language. The framework is flexible and practical to the maintenance of various kinds of products especially to complex products. The detailed work of maintenance planning can be generated automatically according to the correspondence of the concepts between the knowledge category and the business category.

Keywords: knowledge, ontology, product ontology, collaborative maintenance, maintenance planning, maintenance procedure.

1 Introduction

The importance of the maintenance function has increased because of its role in keeping and improving system availability and safety, as well as product quality [1,2]. Collaborative maintenance integrates existing telemaintenance principles, with Web services and modern e-collaboration principles. Collaboration allows to share and exchange not only information but also knowledge and (e)-intelligence [3,4].

Maintenance planning management is one of the most important businesses in product maintenance [5,6] especially to collaborative maintenance. Usually maintenance planning is formulated based on the maintenance policy and maintenance procedure, which is the basis of maintenance scheduling [7,8]. The essential elements for formulating the maintenance planning include the choice of maintenance policy, defining maintenance procedure which is called as maintenance standard by the paper[9] and allocation of maintenance resources. Maintenance activities could not be planned and implemented successfully without full understanding of these elements [7,9].

Ontologies in the Semantic Web can be used to explicitly represent semantics involved when making computer-aided maintenance planning [4,5] in order to promote integrated and consistent access to data and services for collaborative maintenance. It is important to establish a common semantic terminology and frame related to maintenance planning in collaborative maintenance environment.

It is important to establish a common semantic terminology and frame related to maintenance planning in collaborative maintenance environment. The benefits are :(a) to generate assistant maintenance planning more quickly; (b) to estimate maintenance resource requirements such as manpower, working hours, capital and materials more precisely;(c) to describe maintenance experience in a detailed way; (d)to promote the integration and collaboration of different maintenance systems.

This paper is structured as follows: This section introduces the typical mode to make computer-aided maintenance planning and the correlative abstract concepts .Related work is reviewed in Section 2. Section 3 gives one ontology-based framework for collaborative maintenance planning. Using OWL, the model is coded and some scenarios of applying the framework to manage maintenance planning are illustrated in Section 4. Finally the conclusion is given and the future work is discussed in Section 5.

2 Related Work

Little work focused on establishing a common collaborative maintenance semantic terminology and framework. Most researches on collaborative maintenance were mainly concerned about concept analysis [3,10,11], system framework [12,13], or one aspect of collaborative maintenance such as maintenance outsourcing [14,15] .The lack of common maintenance semantic terminology and frame is due to different levels of product complexity and different terminology used in distinct industry domains.

Most researches on maintenance planning concentrated on the choice of maintenance policy or the allocation of maintenance resources. Some researches mainly studied the choice or optimization of maintenance policy [16,17]. Most complex products belong to multi-component systems whose maintenance is complicated. The paper [18] gave a review of multi-component maintenance models with economic dependence. The related researches of optimal maintenance of multi-component systems have been summarized [19] .

The importance of maintenance procedure was recognized widely and discussed by some papers, but little work has considered how to describe and optimize maintenance procedure in detail. The paper [20] discussed the role of 'know-how' in maintenance activities, which is maintenance procedure essentially ,and particularly the problems raised by putting this knowledge into words. Maintenance procedure is also one kind of maintenance instruction essentially. The significance of automating maintenance instructions was discussed in the paper [21]. The paper [22] demonstrated that the studies of the reliability of aircraft inspection and maintenance procedures can have implicated documentation as a contributing factor. The importance of sharing 'know how' to contextual assessment of working practices in changing such as manpower requirements and financial needs was expounded in the paper [23].

An ontology is an explicit and formal specification of a shared conceptualization and provides a conceptual framework for communicating in a given application domain[24].Some papers committed to the sharing of product lifecycle data. Yoo and

Kim[25] presented a Web-based knowledge management system for facilitating seamless sharing of product data among application systems in virtual enterprises. J. Lee etc.[26] suggested one 4-layered ontology architecture for an integrated value chain. The paper[27] develops a novel mechanism for integrating ontology-based product lifecycle knowledge.

Little work proposed an ontology framework to define the concepts related to maintenance planning, although some papers discussed product ontology in the operation and maintenance phases of product lifecycle. More papers focused on studying product ontology in the design and manufacturing phases of product lifecycle[28,29,30]. The paper[31] reported their effort to build an operational product ontology system for a government procurement service. We haven't found valuable research on the models related to maintenance planning.

3 One Ontology-Based Framework for Maintenance Planning

3.1 Analysis of Ontology Models for Maintenance Planning

From the studies of related literatures, we can conclude that the typical mode to make computer-aided maintenance planning is as follow:

If the **values of product status parameters** of one maintenance object meet the judgment conditions of one piece of maintenance policy, **then**

(1) The correlative work of the maintenance type should be done to the **maintenance object**;

(2) The correlative **maintenance planning** could be made according to default **maintenance procedure** which is corresponding to the **maintenance type** of the **maintenance object**;

(3) Maintenance resource including manpower, finance and material could be estimated and allocated.

Seen from the above mode, when making computer-aided maintenance planning there are six crucial factors:

(1)Maintenance object. One maintenance object is commonly one product/component which is called ‘material’ generally. Maintenance objects can be divided into neutral materials and physical materials. One neutral material represents one kind of products/components. One physical material denotes one physical product/component existing in the real world.

(2)Product status parameters. They belong to product maintenance knowledge, which is specific product operating information or environmental data that is needed in maintenance policy such as product faults, product operating conditions.

(3)Maintenance policy. It belongs to product maintenance knowledge that is used to describe in what situations one maintenance activity of one product should be triggered.

(4)Maintenance type. It belongs to product maintenance knowledge, which is the result of the choice of maintenance policy. A maintenance type stands for one level of maintenance work in order to deal with faults, routing maintenance, etc.

(5)Maintenance procedure. It belongs to product maintenance knowledge, which is used to describe how the maintenance work should be accomplished directed when the product status parameter values meet the judgment conditions of maintenance policy.

(6)Maintenance planning. It belongs to product maintenance business data, which is used to enumerate which actual maintenance work will be done and what the detailed maintenance tasks are directed towards the actual exceptional product status parameters.

Seen from the above pivotal factors, we could divide the concepts related to maintenance planning into two categories, namely maintenance knowledge category and maintenance business category. By the same token, the ontology models for maintenance planning could be respectively vested in knowledge category and business category as shown in Fig.1. Generally product lifecycle data are organized based on product structure, which is called product Bill of Materials (BOM) as demonstrated [32,33]. Accordingly, BOM of products/components should be divided into neutral BOM and physical BOM.

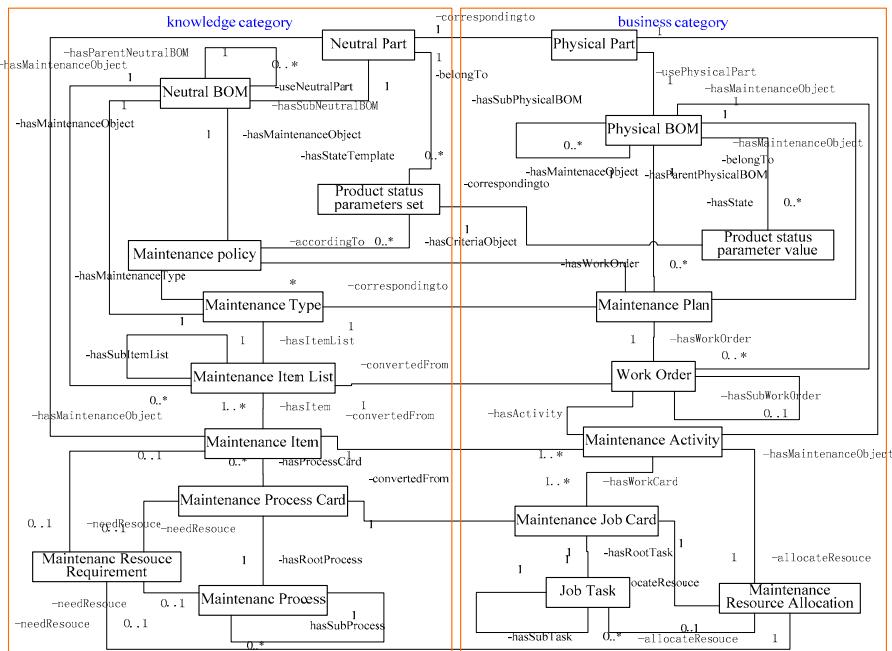


Fig. 1. Ontology models of conceptual layer for making computer-aided maintenance planning

The main challenge of maintenance procedure is how to describe the concrete maintenance process as subtly as possible. For simple product/component or simple maintenance work, the contents of maintenance procedure may be a few simple maintenance items such as refueling, cleaning or lubrication. For complex products/components or overhaul, one maintenance work order is often decomposed into some maintenance sub-work-order or maintenance activities of its specific subparts or itself in fact. What's more, one complex product is composed of some complex components and some simple components. Thus it is necessary to establish a

unified description mechanism of maintenance procedure which is applicable to both simple materials and complex materials. We put forward one recursive presentation framework of maintenance procedure consisting of maintenance item list, maintenance process card and maintenance task as shown in Fig.1.

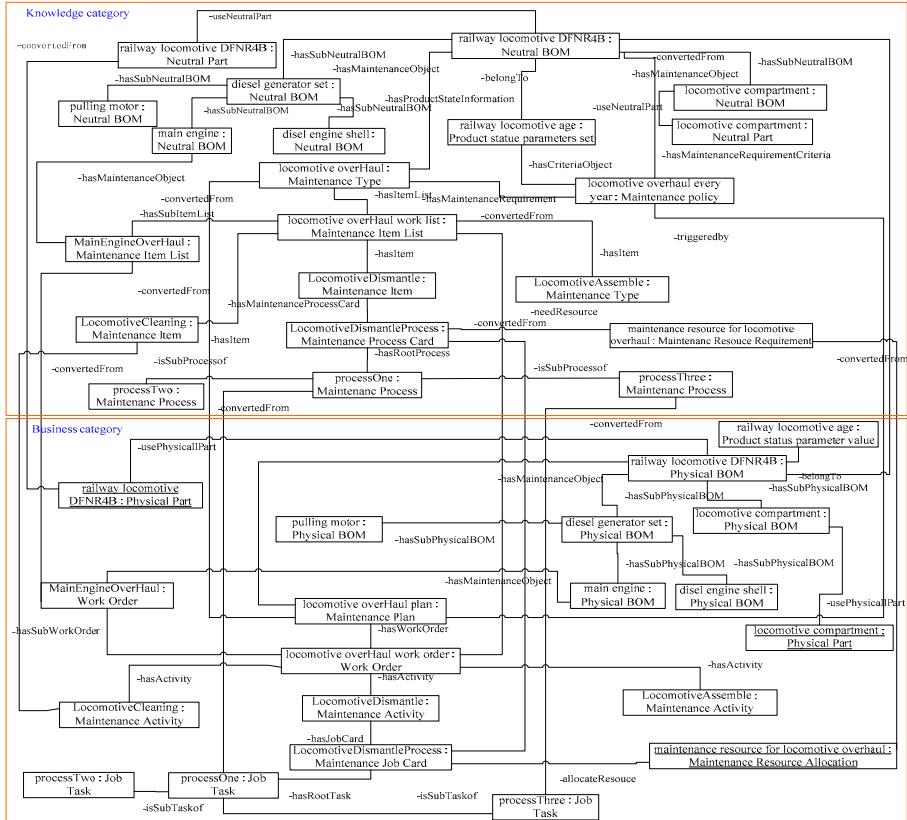


Fig. 2. Ontology models of instance layer for making maintenance planning

The ontology model of maintenance planning has three layers. The first layer is the meta-ontology layer which includes the most abstract model elements includes individuals, properties and classes referenced to ontology modeling methodology of OWL. Individuals represent objects in the domain in which we are interested. Properties are binary relations on individuals. Classes are the sets that contain individuals. The second is the concept layer which shows concrete classes and properties of our product maintenance planning model as shown in Fig.1. The third is the instance layer. Users can manage maintenance knowledge and make specific maintenance planning through creating individuals of the concepts as shown in Fig.2. Then we will explore the concepts and the relationships among them in Fig.1 combined with the example in Fig.2 according to knowledge category and business category respectively.

3.2 Ontology Models in Knowledge Category

The concepts of knowledge category for maintenance planning come from the shared maintenance knowledge which can be reused. The other primary ontology models in maintenance knowledge category are designed as follows except neutral material, neutral BOM, product status parameters and maintenance type.

(1) Maintenance item list. One maintenance item list can contains some maintenance items and some sub-maintenance-item-list. It is to clearly display two aspects of maintenance work in connection with a maintenance type to one neutral material. One is to enumerate which sub-parts should be maintained and what the maintenance type of each of them is. The other is to state which maintenance items of the product itself should be done. So the maintenance item list of one neutral product is the set of both sub-maintenance-item-lists of its sub-parts and maintenance items of itself.

(2) Maintenance item. As explained above, a maintenance item is a finer-grained Maintenance type. When conducting maintenance work, maintenance process knowledge is represented by maintenance process card. As shown in Fig.2, the maintenance item locomotive dismantle has a locomotive dismantle process card.

(3) Maintenance process card and maintenance process. A maintenance process card is a standard route of maintenance work corresponding to one maintenance item of one maintenance object, which is the set of maintenance processes. It ensures that maintenance work is done by standard process. One maintenance activity can be accomplished through distinct technical lines, so one maintenance item can have several maintenance process cards which are alternative. A maintenance process card can have a number maintenance processes which can be further decomposed into many sub-processes. As shown in Fig.2, the railway locomotive maintenance process card is associated with the locomotive dismantle process. In many cases, there are multiply relationships between the maintenance processes of a maintenance process card such as parallel and serial. To simplify it, we just define the sub-process relation between them.

(4) Maintenance resource requirement. It includes manpower, material and tools requirement. To simple materials or simple maintenance types, maintenance process card may be null and maintenance resource requirement is directly linked to maintenance item. To complex materials or overhaul, the maintenance technique route may be very complicated and maintenance resource requirement should be linked to maintenance processes. Considering the flexibility of organizing them, we can associate maintenance item, maintenance process card, or maintenance process with it in order to define larger or smaller ranges to use the resources. As shown in Fig.2, locomotive dismantle work card is associated with a maintenance resource requirement indicating the maintenance resource needed during the whole process of the locomotive dismantle.

3.3 Ontology Models in Maintenance Business Category

The concepts of maintenance business category for maintenance planning come from the actual business requirement. A physical BOM is converted from a neutral BOM, for a physical BOM is an actual copy of neutral BOM but located in different places, composed of different physical materials. So when converted from a Neutral BOM, the Physical Materials that are associated with the physical BOM will be converted from its counterpart of Neutral Materials.

Maintenance plan is a plan of maintenance work with planned time, planned resource allocation, definite maintenance object, work content and instructions. Maintenance plan can be created manually by experience, or, in a more recommendable way, be created according to maintenance policy. Once the criterion is satisfied, the maintenance plan will be converted from the maintenance type that associated with the criterion, and definite time planning is added to the plan. As exemplified in Fig.2, maintenance policy of railway locomotive overhaul every year produces a periodical plan of overhaul upon the railway locomotive. The other ontology models in maintenance procedures management are designed as follows except the concepts of physical material, physical BOM, product status parameter values and maintenance plan.

(1)Work order. Work order is a definite set of actual maintenance work. It can be converted from one maintenance item list. A Work order can has a sub-work-order which involves the maintenance work that should be done on sub-physical-BOM at the same time, and it has maintenance activity to further decompose the maintenance work instruction. As depicted in Fig.2, the wok order of railway locomotive overhaul is created one year after the last overhaul of railway locomotive, and it has sub-work-order of main engine overhaul. It has maintenance activity of locomotive dismantle, locomotive cleaning and so on.

(2)Maintenance activity. It further decomposes work order into many kinds of wok instruction. It has several job cards from which people could choose its concrete work procedure. Maintenance activity is converted from maintenance item with the corresponding maintenance item list. As depicted in Fig.2, the maintenance activity of locomotive dismantle is converted from its counterpart of maintenance item.

(3)Job card. A job card defines certain work process of doing a specific maintenance activity. It can be generated automatically based one of corresponding maintenance process card.

(4)Job task. It is the most basic step to instruct maintenance work. A job card has some job tasks that can be further decomposed into sub-maintenance-job-tasks. As shown in Fig.2, the job card of locomotive dismantle has the maintenance job task of locomotive dismantle task.

(5)Maintenance resource allocation. Maintenance resource allocation is real and concrete allocation of specific labor power, materials and tools with which to complete actual maintenance work. Just as the considerations of maintenance resource requirement, maintenance resource allocation can be associated with job task, job card, maintenance activity or work order.

4 Modeling of the Concepts of Maintenance Planning Using Ontology Language

We have establish the ontology models of all the above concepts in Protégé which encoded in OWL. With the syntax of RDF/XML, the concepts and the relationships among them for making maintenance planning can be understood by both machines and people. The correspondence among them is shown in table I. When creating one maintenance planning, the assistant detailed work of maintenance planning for one physical product/component can be generated according to the correspondence of the concepts between knowledge category and business category. The standard work of one maintenance type to one neutral product/component can be improved based on

mass actual business data of its many corresponding physical materials. In collaborative maintenance environment, most work of maintenance planning such as maintenance activities, maintenance sub-work-orders, job cards and even job tasks can be accomplished by different maintenance service providers.

Table 1. The correspondence of ontology models between knowledge category and business category

Ontology models in knowledge category	Ontology models in business category
neutral BOM	Physical BOM
product status parameter set	Product status parameter values
Maintenance type	Maintenance plan
Maintenance item list	Work order
Maintenance item	Maintenance activity
Maintenance process card	Job card
Maintenance process	Job task
Maintenance resource requirement	Maintenance resource allocation

Taking the overhaul work of railway locomotive DFNR4B and corresponding maintenance concepts as an example, the detailed definitions can be seen in Fig. 3. The railway locomotive overhaul is one maintenance type. Railway locomotive DFNR4B is one neutral material.

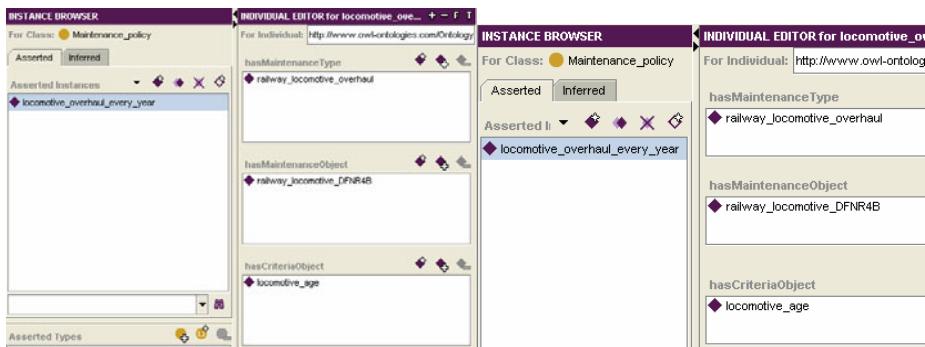


Fig. 3. Railway locomotive overhaul and corresponding maintenance concepts

5 Conclusions

In this paper, one ontology-based framework for collaborative maintenance planning in collaborative maintenance environment is proposed. We have applied the above framework in one coal mining equipment maintenance management system. Through maintenance item lists and maintenance processes which can be broken down into multilayer sub-working procedures, the framework is flexible and applicable to the business demands of various kinds of products, especially to complex products. The assistant detailed work contents of maintenance planning for one physical material can be generated according to the correspondence of the concepts between knowledge category and business category. In collaborative maintenance environment, most work contents of maintenance planning can be accomplished by different maintenance service providers.

But we haven't considered some important and common business demands in the above framework. For example, one kind of complex product may have some components which can be replaced with each other and multiple design versions in which different versions have different instance business object. Our work mainly focuses on how to describe the concrete maintenance steps of the maintenance planning as detailedly as possible. The unified ontology-based framework for maintenance policy is also one great challenge. The future work should focus on these issues.

Acknowledgments. Authors of this paper would like to thank Chinese 863 Planning. This research was financially upheld by Project 2009AA043401 and 2009AA043402 supported by Chinese 863 Planning.

References

1. Khan, M.R.R., Darrab, I.A.: Development of Analytical Relation between Maintenance, Quality and Productivity. *Journal of Quality in Maintenance Engineering* 16(4), 341–353 (2010)
2. Alsyouf, I.: The Role of Maintenance in Improving Companies' Productivity and Profitability. *International Journal of Production Economics* 105(1), 70–78 (2007)
3. Muller, A., Crespo Marquez, A., Iung, B.: On the Concept of E-Maintenance: Review and Current Research. *Reliability Engineering & System Safety* 93(8), 1165–1187 (2008)
4. Iung, B., Levrat, E., Marquez, A.C., Erbe, H.: Conceptual Framework for E-Maintenance: Illustration by E-Maintenance Technologies and Platforms. *Annual Reviews in Control* 33(2), 220–229 (2009)
5. Garg, A., Deshmukh, S.: Maintenance Management: Literature Review and Directions. *Journal of Quality in Maintenance Engineering* 12(3), 205–238 (2006)
6. Dhillon, B., Liu, Y.: Human Error in Maintenance: A Review. *Journal of Quality in Maintenance Engineering* 12(1), 21–36 (2006)
7. Yang, Z.M., Djurdjanovic, D., Ni, J.: Maintenance Scheduling in Manufacturing Systems Based on Predicted Machine Degradation. *Journal of Intelligent Manufacturing* 19(1), 87–98 (2008)
8. Garver, L.: Adjusting Maintenance Schedules to Levelize Risk. *IEEE Transactions on Power Apparatus and Systems* (5), 2057–2063 (2007)
9. Lee, H.H.Y., Scott, D.: Overview of Maintenance Strategy, Acceptable Maintenance Standard and Resources from a Building Maintenance Operation Perspective. *Journal of Building Appraisal* 4(4), 269–278 (2009)
10. Iung, B.: From Remote Maintenance to Mas-Based E-Maintenance of an Industrial Process. *Journal of Intelligent Manufacturing* 14(1), 59–82 (2003)
11. Lee, J., Ni, J., Djurdjanovic, D., Qiu, H., Liao, H.: Intelligent Prognostics Tools and E-Maintenance. *Computers in Industry* 57(6), 476–489 (2006)
12. Trappey, A.J.C., Hsiao, D.W., Ma, L., Chung, Y.L., Kuo, Y.L.: Agent-Based Collaborative Maintenance Chain for Engineering Asset Management. *Collaborative Product and Service Life Cycle Management for a Sustainable World*, 29–41 (2008)
13. Bangemann, T., Rebeuf, X., Reboul, D., Schulze, A., Szymanski, J., Thomesse, J.P., Thron, M., Zerhouni, N.: Proteus—Creating Distributed Maintenance Systems through an Integration Platform. *Computers in Industry* 57(6), 539–551 (2006)
14. Van Niekerk, A., Visser, J.: The Role of Relationship Management in the Successful Outsourcing of Maintenance. *South African Journal of Industrial Engineering* 21(2), 79–90 (2010)

15. Persona, A., Regattieri, A., Pham, H., Battini, D.: Remote Control and Maintenance Outsourcing Networks and Its Applications in Supply Chain Management. *Journal of Operations Management* 25(6), 1275–1291 (2007)
16. Castanier, B., Grall, A., Berenguer, C.: A Condition-Based Maintenance Policy with Non-Periodic Inspections for a Two-Unit Series System. *Reliability Engineering & System Safety* 87(1), 109–120 (2005)
17. Laggoune, R., Chateauneuf, A., Aissani, D.: Opportunistic Policy for Optimal Preventive Maintenance of a Multi-Component System in Continuous Operating Units. *Computers & Chemical Engineering* 33(9), 1499–1510 (2009)
18. Dekker, R., Wildeman, R.E., van der Duyn Schouten, F.A.: A Review of Multi-Component Maintenance Models with Economic Dependence. *Mathematical Methods of Operations Research* 45(3), 411–435 (1997)
19. Nicolai, R.P., Dekker, R.: Optimal Maintenance of Multi-Component Systems: A Review. In: *Complex System Maintenance Handbook*, pp. 263–286 (2008)
20. Garrigou, A., Carballeda, G., Daniellou, F.: The Role of [] Know-How'in Maintenance Activities and Reliability in a High-Risk Process Control Plant. *Applied Ergonomics* 29(2), 127–131 (1998)
21. Badler, N., Erignac, C., Vincent, P., Sanchez, E., Boyle, E.S.: Design Concepts for Automating Maintenance Instructions, Defense Technical Information Center, P. U. Philadelphia (2000)
22. Chervak, S.G., Drury, C.G.: Effects of Job Instruction on Maintenance Task Performance. *Occupational Ergonomics* 3(2), 121–131 (2003)
23. Nuutinen, M.: Contextual Assessment of Working Practices in Changing Work. *International Journal of Industrial Ergonomics* 35(10), 905–930 (2005)
24. Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition* 5, 199 (1993)
25. Yoo, S.B., Kim, Y.: Web-Based Knowledge Management for Sharing Product Data in Virtual Enterprises. *International Journal of Production Economics* 75(1-2), 173–183 (2002)
26. Lee, J., Chae, H., Kim, C.H., Kim, K.: Design of Product Ontology Architecture for Collaborative Enterprises. *Expert Systems with Applications* 36(2), 2300–2309 (2009)
27. Chen, Y.J., Chen, Y.M., Chu, H.C.: Development of a Mechanism for Ontology-Based Product Lifecycle Knowledge Integration. *Expert Systems with Applications* 36(2), 2759–2779 (2009)
28. Dong, M., Yang, D., Su, L.: Ontology-Based Service Product Configuration System Modeling and Development. *Expert Systems with Applications* (2011)
29. Gimenez, D.M., Vegetti, M., Leone, H.P., Henning, G.P.: Product Ontology: Defining Product-Related Concepts for Logistics Planning Activities. *Computers in Industry* 59(2-3), 231–241 (2008)
30. Yang, D., Dong, M., Miao, R.: Development of a Product Configuration System with an Ontology-Based Approach. *Computer-Aided Design* 40(8), 863–878 (2008)
31. Lee, T., Lee, I., Lee, S., Kim, D., Chun, J., Lee, H., Shim, J.: Building an Operational Product Ontology System. *Electronic Commerce Research and Applications* 5(1), 16–28 (2006)
32. Wu, M.C., Hsu, Y.K.: Design of Bom Configuration for Reducing Spare Parts Logistic Costs. *Expert Systems with Applications* 34(4), 2417–2423 (2008)
33. Jian-guo, Y., Zhang, G., Pei-gen, L.: The Key Technology Research of Bom-Oriented Manufacturing Information System Integration. *Application Research of Computers* 4 (2004)

A Framework for a Fuzzy Matching between Multiple Domain Ontologies

Konstantin Todorov¹, Peter Geibel², and Céline Hudelot¹

¹ Laboratory MAS, École Centrale Paris

² Fakultät IV, TU Berlin

Abstract. The paper proposes an alignment framework for a set of domain ontologies in order to enable their interoperability in a number of information retrieval tasks. The procedure starts by anchoring the domain ontologies concepts to the concepts of a generic reference ontology. This allows the representation of each domain concept as a fuzzy set of reference concepts or instances. Next, the domain concepts are mapped to one another by using fuzzy sets relatedness criteria. The match itself is presented as a fuzzy set of the reference concepts or instances, which allows the comparison of a new ontology directly to the already calculated matches. The paper contains a preliminary evaluation of the approach.

1 Introduction

With the growing demand and acceptance of ontology-based applications, we have witnessed the creation of multiple ontologies describing similar or even identical fractions of real world knowledge. These ontologies, (partly) complementary or (partly) redundant, have an impaired collaborative functionality, because of the decentralized nature of their conception, their different scopes and application purposes, or because of mismatches in terms of syntax and terminology. More than rarely, however, the sharing, integration and interoperability of these resources is required in real life application scenarios.

Ontology matching provides mechanisms for the alignment of (the components of) various knowledge resources. The different ontology matching approaches can be classified w.r.t. the object on which this alignment relies [10]: *terminological approaches* measure the similarity of the concept names and their lexical definitions, *extensional* approaches use instance data to discover matches, *structural* approaches rely on the relations that hold between the different concepts and *semantic* approaches are based on logical methods. These different approaches are often complemented by the use of background knowledge provided by a *reference ontology*, allowing to deal with realistic matching cases (e.g. weakly structured models) [2][16][15]. Another current issue in realistic case ontology matching is the handling of imprecise information and the resulting matching imperfections [12].

The paper suggest a procedure for alignment of the concepts of several domain ontologies, referred to as source ontologies, by the help of a generic reference ontology. The reference ontology is a pre-existing knowledge body which provides

common knowledge about a given domain of interest. The choice of a reference therefore depends on the sources: this can be a broader domain ontology (for instance FMA in the domain of medicine or biology) or a more generic knowledge source (such as populated WordNet or Wikipedia). In the current study, we focus on text-populated hierarchies (e.g. web-directories) describing similar or complementary domains, using Wikipedia as a reference ontology. We apply the idea of anchoring a set of source ontologies onto a reference ontology [15], but in contrast to previous techniques, we rely on the uniformity in the matching criteria for the whole set of input ontologies as well as the semantic nature of these matchings as a main advantage of the anchoring. Based on this anchoring, we redefine the source concepts as fuzzy sets of reference concepts or, consequently, instances. This enables the application of a whole set of similarity measures defined on fuzzy sets. An important and difficult question is how a concept is defined, how many and what instances are included in its extension. This uncertainty in concept definition is embedded by entering the realm of fuzzy set representations. In consequence, uncertainty in concept matching is addressed as well. The match itself is presented as a fuzzy set of the reference concepts or instances, which plays in favor of the scalability of the approach.

As we shall see in the sequel, certain analogies between our approach and topic modeling [5] can be drawn. Our results particularly relate to the LDA approach taken by Rosen-Zvi *et al.* [17], who determine the similarity of authors based on topic vectors which describe the respective authors publications. In contrast to their approach, our design decision was to use pre-existing knowledge in the form of a reference ontology for the topics. After computing the topic scores for the source ontologies, our approach is able to compute new “topic models” for the matches, without using the instances any more.

In next section, we discuss related work. Section 3 provides background in the problem of ontology heterogeneity and describes standard measures for extensional concept mapping, as well as a novel ontology matching algorithm. The framework of the alignment approach that we propose is presented in Section 4 followed by a preliminary evaluation in Section 5.

2 Background and Related Work

Fuzzy set theory has been introduced as a generalization of classical set theory [22]. A fuzzy set A is defined on a given domain of objects X by the function f_A which expresses the degree of membership of every element of X to A by assigning to each $x \in X$ a value from the interval $[0, 1]$. This allows to deal with imprecise and vague data. A way of handling imprecise information in ontologies is to incorporate fuzzy logic into them. Several papers by Sanchez, Calegari and colleagues [7], [8], [18] form an important body of work on fuzzy ontologies. The authors have been motivated by the observation that crisp reasoning through two valued logic, although machine processable, is not suited to deal with uncertain or imprecise information available in real world knowledge. Each ontology concept is defined as a fuzzy set on the domain of instances and relations on the domain of instances and concepts are defined as fuzzy relations.

Work on fuzzy ontology matching can be classified in two families : (1) approaches extending crisp ontology matching to deal with fuzzy ontologies and (2) approaches addressing imprecision of the matching of (crisp or fuzzy) concepts. Based on the work on approximate concept mapping by Stuckenschmidt [19] and Akahani *et al.* [1], Xu *et al.* [21] suggested a framework for the mapping of fuzzy concepts between fuzzy ontologies. With a similar idea, [4] propose a framework to define similarity relations among fuzzy ontology components. The other family of fuzzy matching approaches is motivated by the representation of imprecision of the matching itself, even with crisp ontologies. For instance, in [11], a fuzzy approach is proposed to handling mapping uncertainty. A new ontology mapping approach based on fuzzy conceptual graphs and rules is proposed in [6]. To define new intra-ontology concept similarity measures, Cross *et al.* [9] model a concept as a fuzzy set of its ancestor concepts and itself. As a membership degree function, the authors use the Information Content (IC) of concept with respect to its ontology. IC can be measured by using external text corpus or by using the ontology structure.

3 Matching Heterogeneous Ontologies

An ontology consists of a set of *concepts* and *relations* defined on these concepts, which provide in an explicit and formal manner knowledge about a given domain. We are particularly interested in ontologies, whose concepts come equipped with a set of associated instances, referred to as populated ontologies and defined as tuples of the kind $O = \{C, \text{is_a}, R, I, g\}$, where C is a set whose elements are called concepts, is_a is a partial order on C , R is a set of other relations holding between the elements of C , I is a set whose elements are called instances and $g : C \rightarrow 2^I$ is a mapping from the set of concepts to the set of subsets of I . In this way, a concept is *intensionally* modeled by its relations to other concepts, and *extensionally* by a set of instances assigned to it via the mapping g . By assumption, all instances can be represented as real-valued vectors of uniform dimension.

Ontology heterogeneity occurs when two or more ontologies are created independently from one another over similar domains. Heterogeneity may be observed on a *linguistic or terminological* level (use of vocabulary), on a *conceptual* level (level of detail, coverage or scope) [10] or on *extensional* level (population). Whenever heterogeneity of any of these kinds is observed over a set of ontologies, these ontologies will be referred to as *heterogeneous*.

Ontology matching addresses the heterogeneity problem by providing a set of assertions on the relations holding between the elements of two (or more) heterogeneous ontologies. In a narrower understanding of this definition, we will be interested in measuring the degree of equivalence of any two concepts from two distinct ontologies. Under a given choice of similarity criteria, various measures of concept relatedness can be applied. We have proposed several extensional concept similarity measures for document populated ontologies in [20].

Let us consider two ontologies $O_1 = (C_1, \text{is_a}, R_1, I_1, g_1)$ and $O_2 = (C_2, \text{is_a}, R_2, I_2, g_2)$. For the purposes of the current study, we have relied on the straightforward idea that determining the similarity $\text{sim}(A, B)$ of two concepts $A \in C_1$ and $B \in C_2$ consists in comparing their instance sets $g_1(A)$ and $g_2(B)$. For doing so, we need a similarity measure for instances \mathbf{i}^A and \mathbf{i}^B , where $\mathbf{i}^A \in g_1(A)$ and $\mathbf{i}^B \in g_2(B)$. We can use, for instance, the scalar product and the cosine $s(\mathbf{i}^A, \mathbf{i}^B) = \frac{\langle \mathbf{i}^A, \mathbf{i}^B \rangle}{\|\mathbf{i}^A\| \|\mathbf{i}^B\|}$. Based on this similarity measure for elements, the similarity measure for the sets can be defined by computing the similarity of the mean vectors corresponding to class prototypes, i.e.

$$\text{sim}_{proto}(A, B) = s\left(\frac{1}{|g_1(A)|} \sum_{j=1}^{|g_1(A)|} \mathbf{i}_j^A, \frac{1}{|g_2(B)|} \sum_{k=1}^{|g_2(B)|} \mathbf{i}_k^B\right). \quad (1)$$

This method underlies the CAIMAN approach [14] in which concepts are assumed to be represented by their mean vector. The theory of hierarchical clustering (e.g., [3]) provides alternative methods for defining similarities for pairs of sets. Examples are the similarity measures sim_{min} , sim_{max} , sim_{avg} , which use the minimum, maximum, and average similarity of concept vectors, respectively.

Using the prototype corresponds to developing a simple topic model, in which the topics correspond to the word weights in the prototype. More elaborate topic modelling like LDA [5] and PLSI [13] could be applied, which are able to determine the underlying, “hidden” topics of the documents in a concept. However, in our experiments, using the prototype vector already worked well. It is also computationally much less expensive, than, for instance, PLSI. Note that our fuzzy approach, to be introduced later, consists in **providing** a hierarchical set of topics in the form of a reference ontology. The semantics of the reference topics is described by their instances.

A *matching algorithm* based on a concept similarity measure like one of the suggested above, is given in Alg. II. The algorithm operates implicitly on the product graph of two input ontologies and it takes into account all given relations in these ontologies (the **is_a** relation and the relations in R are treated in the same manner). The algorithm has a quadratic runtime, since it is greedy and operates on the product graph.

4 A Two-Level Multiple Ontologies Matching Architecture

Let $\Omega = \{O_1, \dots, O_n\}$ be a set of ontologies that will be referred to as the set of *source* ontologies and let their concepts be referred to as *source concepts*, denoted by C_Ω . Let $O_{ref} = (X, \text{is_a}, R_{ref}, I_{ref}, g_{ref})$, be an ontology, called the *reference ontology* whose concepts will be called *reference concepts*. The set Ω is characterized as a set of ontologies which share similar functionalities and application focuses, but are heterogeneous as discussed in Section B. A certain complementarity of these ontologies can be assumed: they could be defined with the same application scope, but on different levels, treating different and complementary aspects of the same

```

procedure Map( $M, O_1, O_2$ )
  //  $M$  is the set of matches that are still possible
  // The function returns a set of mappings for the concepts of  $O_1$  and  $O_2$ 
begin
  | Find  $(A, B) \in M$  that maximizes  $sim(A, B)$ 
  | // The parameter  $\theta$  specifies the acceptable minimum similarity
  | If  $sim(A, B) < \theta$  then return  $\emptyset$ 
  | // Because of the ISA-relationships and the ones in  $R$ , the match  $(A, B)$ 
  | // constrains the remaining set of potential matches:
  | Remove the following from  $M$ :
    | – every pair  $(a, b)$  such that one of the following conditions is true for some
      |  $r \in (R_1 \cap R_2) \cup \{isa\}$ :
        |  $r(a, A) \wedge \neg r(b, B), r(b, B) \wedge \neg r(a, A), r(A, a) \wedge \neg r(B, b), r(B, b) \wedge \neg r(A, a)$ 
    | – every pair  $(a, b)$  for which  $A = a, B = b$  //in order to enforce a 1:1 mapping
  | return  $\{(A, B)\} \cup Map(M, O_1, O_2)$ 
end

procedure Main( $O_1, O_2$ )
begin
| return Map( $C_1 \times C_2, O_1, O_2$ )
end

```

Algorithm 1. A greedy algorithm for matching ontologies O_1 and O_2

application problem. The ontology O_{ref} is assumed to be application independent, generic knowledge source. Finally, we posit that the ontologies in Ω and O_{ref} are populated as described in section 3. We are interested in identifying the degree of relatedness of any two concepts taken from any two ontologies from the set Ω . We propose the following matching architecture.

Phase one. The source ontologies are first matched independently from one another to the reference ontology by the help of the concept similarity measures and the algorithm introduced in Section 3. As a result, every concept from each of the source ontologies can be represented as a set of similarity scores calculated for this concept and all the concepts in the reference ontology (the *score* in our case is one of the *sim* functions introduced in the previous section).

The considered concept representation gives rise to the following fuzzy set interpretations. Let $score_A(x)$ be the similarity between a concept $x \in X$ and A , a random concept from the set of source ontologies. The concept A will be defined as a fuzzy set in X which has a membership function f_A given by

$$f_A(x) = score_A(x), \forall x \in X. \quad (2)$$

Alternatively, we propose to define the membership function on the set of *instances* of the reference ontology concepts. We will be looking for a function of some domain element, i , in A which maximizes the scores of those concepts in the reference ontology that contain i as an instance. We start by presenting the reference ontology as an ontology of fuzzy concepts with respect to its instances.

In our case, this will be trivialized to a two-valued membership function: for every instance \mathbf{i} from the reference ontology the function $f_x(\mathbf{i}) = 1$, if $\mathbf{i} \in x$ and 0 otherwise. Thus, we define a source concept A as

$$f_A(\mathbf{i}) = \max_{x \in X} T(f_x(\mathbf{i}), f_A(x)), \quad (3)$$

where T is the t -norm of two fuzzy membership functions defined as $T(f_A, f_B) = \min(f_A, f_B)$. Recall that in fuzzy set theory the t -norm and the t -conorm (defined as $S(f_A, f_B) = \max(f_A, f_B)$) carry the sense of intersection and union of fuzzy sets. As required, (3) amounts to finding the concept x that contains the instance \mathbf{i} and has the maximum score with respect to A . We note that this formulation can be potentially extended to fuzzy reference ontologies with standard membership functions in the full interval $[0, 1]$.

Phase two. We will rely on the fuzzified versions of the concepts of the source ontologies in order to judge on their relatedness. Consider two concepts A and B defined by their fuzzy membership functions f_A and f_B . A straightforward measure of the closeness of these concepts can be given as $\rho_{base}(f_A, f_B) = \max_{x \in X} |f_A(x) - f_B(x)|$ or, alternatively, by their Euclidean distance:

$$\rho_{diff}(f_A, f_B) = \|f_A - f_B\|_2, \quad (4)$$

where $\|x\|_2 = (\sum_{x \in X} |x|^2)^{1/2}$ is the l^2 -norm.

Many measures of fuzzy set compatibility known from fuzzy set theory can be applied, as well [9]. Zadeh's partial matching index between two fuzzy sets A and B is given by $\rho_{sup-min}(f_A, f_B) = \sup \min_{x \in X} (f_A(x), f_B(x))$. We also consider the standard Jaccard coefficient $\rho_{jacc}(f_A, f_B) = \#T(f_A, f_B)/\#S(f_A, f_B)$, where $\#$ returns fuzzy set cardinality.

Once we have represented our source concepts as fuzzy sets, we can measure concept similarities directly on the set of fuzzified concepts C_Ω . Alternatively, in order to take the semantical structure of the ontologies into account, one can apply the matching algorithm described in Alg. 1 by taking as input any two given fuzzified source ontologies and using one of the similarity measures introduced above. Fuzzufied versions of the relationships can be used in a modified version of the algorithm, as well, but this remains out of the scope of this paper.

Finally, note that it is possible to define the match itself as a fuzzy set on the reference concepts or their instances (alternative choices given definitions 2 and 3). This will play in favour of the scalability of our approach, since the concepts of every new ontology can be compared to the match directly. One natural possibility of defining the match would be to use again the t -norm. If we know that a source concept A is mapped to a source concept B (information that is made available by the measures introduced earlier in this section), the match will be defined as the fuzzy set

$$f'_{(A,B)}(x) = T(f_A(x), f_B(x)), \forall x \in X. \quad (5)$$

We can easily compare the concepts of a new ontology, represented as well as fuzzy sets on the same space X , with the calculated matches.

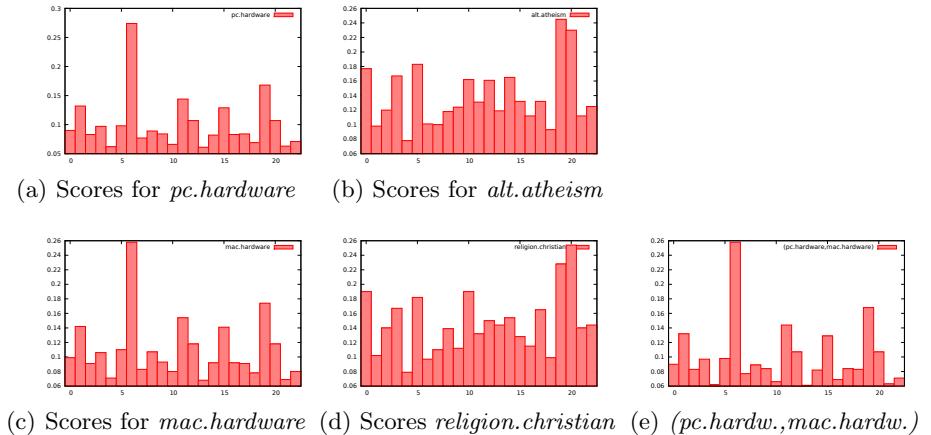


Fig. 1. Fuzzy membership functions: Scores w.r.t. the Inex 2007 Wikipedia Ontology. (a)–(d) represent single concept scores, while (e) represents the scores of the match of two concepts.

5 Experiments

We provide a preliminary evaluation of the proposed approach as a proof of concept. Note that the current section does not aim at comparing the approach to other matching techniques, as this has been done in [20]. It aims to prove, by performing these matchings, that the transition to a fuzzy framework is successful.

As a reference ontology, we consider the 23 categories that form Wikipedia's main topic classifications. For each topic category, we included a set of matching documents from the Inex 2007 corpus which directly belong to this category, or to one of its direct subcategories in the Wikipedia category tree. Thus we arrived at the following 23 concepts: *law* (745 documents), *technology* (293), *arts*(319), *society*(2050), *agriculture*(530) *social_sciences*(1695) *computing*(1902) *health*(341) *education*(515) *mathematics*(1903) *people*(136) *business*(1202) *science*(547), *history*(445), *politics*(896), *applied_sciences*(1302), *geography*(164), *chronology*(303), *environment*(467), *nature*(234), *humanities*(537), *language*(427), *culture*(765). Note that there exist *is_a* relationships between some of the concepts, e.g., *politics* and *society*.

The two source ontologies were constructed from the 20 Newsgroup dataset and consist of the following hierarchically organized classes: $O_1 = \{sci.med\ (990), rec.autos\ (990), alt.atheism\ (799), sport.baseball\ (994), pc.hardware\ (982)\}$ and $O_2 = \{sci.space\ (987), rec.motorcycles\ (993.7), religion.christian\ (997), sport.hockey\ (999), mac.hardware\ (961)\}$.

By applying the techniques described in Section 3, we match these concepts to the reference ontology in order to acquire their fuzzy representations. We then proceed to apply the similarity measures suggested in Section 4 on the set of source concepts. Finally, we also measure their similarity by using the crisp

A from O_1	B from O_2	ρ_{diff}	A from O_1	B from O_2	sim_{proto}
sci.med	sci.space	0.23	sci.med	religion.christ.	0.359
rec.autos	rec.motorcycl.	0.173	rec.autos	rec.motorcycl.	0.471
alt.atheism	religion.christ.	0.078	alt.atheism	religion.christ.	0.537
sport.baseball	sport.hockey	0.068	sport.baseb.	sport.hockey	0.559
pc.hardware	mac.hardware	0.05	pc.hardware	mac.hardware	0.716

Fig. 2. (a) Fuzzy match determined by smallest distances ρ_{diff} . (b) Crips match determined using largest similarities sim_{proto} (larger values are better)

matching measure used in the first step and compare the results achieved by both. In what follows, we will focus on instance-based concept similarities.

In the crisp matching step and also for directly matching O_1 and O_2 , we considered sim_{proto} , sim_{min} , sim_{max} , and sim_{avg} . Since the prototype method worked best and is the most efficient to compute, we will only present the results for this method. In order to compute the prototype similarity, we first transformed the documents into TF-IDF vectors. The prototype method then computes a single prototype (mean vector) for each class. For a pair of classes, their similarity corresponds to the cosine of their prototype vectors.

The diagrams in Fig. 1 show the scores with respect to the Inex 2007 Wikipedia ontology. It can be seen that the membership functions of *pc.hardware* and *mac.hardware* are quite similar, as are those of *alt.atheism* and *religion.christian*. In contrast, *alt.atheism* and *religion.christian* are quite dissimilar to the hardware classes. The two religion-related concepts have their two highest peaks at the Wikipedia concepts *humanities* and *nature*. For the two hardware classes, the Wikipedia concept with the highest score is *computing*.

Using the Euclidean distance $\rho_{diff}(f_A, f_B)$ for selecting the best-matching concept pairs in O_1 and O_2 , we arrive at the match in Fig. 2(a). The fuzzy matching method is obviously able to map the related yet different concept pairs. Even the less obvious match between *sci.med* and *sci.space* is found by the method. Note that it is possible to define a fuzzy membership function of the matched concept (*computers1,computers2*), which is obtained as the minimum of the respective scores. The one for the match (*pc.hardware,mac.hardware*) is shown in Fig. 1(e). Fig. 2(b) shows the match that is found by comparing the prototypes of the respective concepts (higher values are better), i.e., the result of the crisp match between O_1 and O_2 . The match is quite similar to the fuzzy one which proves the correctness of the latter. The crisp method fails to map *sci.med* to *sci.space*.

Mathematically speaking, both the fuzzy matching approach and the prototype approach describe each concept by a single feature vector that is somehow obtained from the document vectors. The one for the fuzzy method corresponds to the scores with respect to the reference concepts, whereas the concept prototype is the average of the document vectors. However, the concept prototype refers to the document *word content* only, whereas the membership function refers to the reference concepts that can be assumed to be of a more *semantic* nature, and to contain only relevant information.

Our fuzzy approach for comparing two concepts scales well, since each concept is described by a number of scores, i.e., a single vector whose length only depends on the number of concepts in the reference ontology, which is assumed to be fixed. In the experiments, we based the scores on the concept prototypes, which is a simple yet efficient method. Using other distances like single, complete and average link models or similarities based on variable selection will result in a higher complexity, but potentially more accurate results. The complexity of the matching algorithm depends on the densities of the graphs and the setting of θ . It can be reduced by forcing the algorithm to descend on the `is_a` relationship, similar to the levelwise algorithm described in [20].

6 Conclusion and Future Work

We have proposed a technique for alignment of the concepts of a set of domain ontologies by using a fuzzy set formulation and a generic reference ontology as a mediator. Fuzziness helps to embed uncertainty in concept definition and representation while the use of a reference ontology provides uniform semantic criteria for this representation. The computation of the match itself is inexpensive.

The suggested approach consists in a change of perspective: we enter the realm of fuzzy reasoning, in which we do not have to use the documents any more. In future work, we will be investigating the idea of building a combined knowledge body on the basis of the redefined fuzzy concepts (taken from the whole set of source ontologies) by exploring the possible fuzzy relations between them, instead of using a pairwise concept matching approach.

In contrast to approaches from the topic modeling theory, in our framework the topic space is defined in the very beginning by the reference ontology; we do not try to induce it from corpora by discovering hidden semantics, but we have clearly defined topics, which are not only word probabilities. This is a different perspective which has as an advantage that depending on the domain of interest, different semantics can be considered by the user with respect to the choice of a reference ontology, i.e. the user can introduce certain bias independent on the latent semantical contents of the instances. In future work, it would be interesting to explore, given a set of source ontologies, how the matching results will differ with respect to different choices of a reference ontology.

References

1. Akahani, J.-I., Hiramatsu, K., Satoh, T.: Approximate query reformulation based on hierarchical ontology mapping. In: Proc. of Intl. Workshop on SWFAT, pp. 43–46 (2003)
2. Aleksovski, Z., Klein, M., Ten Kate, W., Van Harmelen, F.: Matching unstructured vocabularies using a background ontology. In: Managing Knowledge in a World of Networks, pp. 182–197 (2006)
3. Alpaydin, E.: Introduction to Machine Learning. The MIT Press, Cambridge (2004)

4. Bahri, A., Bouaziz, R., Gargouri, F.: Dealing with similarity relations in fuzzy ontologies. In: IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2007, pp. 1–6. IEEE, Los Alamitos (2007)
5. Blei, D., Ng, A., Jordan, M.: Latent dirichlet allocation. *J. Mach. Learn. Res.* 3, 993–1022 (2003)
6. Buche, P., Dibie-Barthélemy, J., Ibanescu, L.: Ontology mapping using fuzzy conceptual graphs and rules. In: ICCS Supplement, pp. 17–24 (2008)
7. Calegari, S., Ciucci, D.: Fuzzy ontology, fuzzy description logics and fuzzy-owl. In: Masulli, F., Mitra, S., Pasi, G. (eds.) WILF 2007. LNCS (LNAI), vol. 4578, pp. 118–126. Springer, Heidelberg (2007)
8. Calegari, S., Sanchez, E.: A fuzzy ontology-approach to improve semantic information retrieval. In: URSW (2007)
9. Cross, V., Yu, X.: A fuzzy set framework for ontological similarity measures. In: WCCI 2010, FUZZ-IEEE 2010, pp. 1–8. IEEE Computer Society Press, Los Alamitos (2010)
10. Euzenat, J., Shvaiko, P.: *Ontology Matching*, 1st edn. Springer, Heidelberg (2007)
11. Ferrara, A., Lorusso, D., Stamou, G., Stoilos, G., Tzouvaras, V., Venetis, T.: Resolution of conflicts among ontology mappings: a fuzzy approach. In: OM 2008 at ISWC (2008)
12. Gal, A., Shvaiko, P.: Advances in Ontology Matching. In: Advances in web semantics i, pp. 176–198. Springer, Heidelberg (2009)
13. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1999, pp. 50–57. ACM, New York (1999)
14. Lacher, M.S., Groh, G.: Facilitating the exchange of explicit knowledge through ontology mappings. In: Proceedings of the 14th FLAIRS Conf., pp. 305–309. AAAI Press, Menlo Park (2001)
15. Noy, N., Musen, M.: Anchor-prompt: Using non-local context for semantic matching. In: Workshop on Ontologies and Information Sharing at IJCAI, pp. 63–70 (2001)
16. Reynaud, C., Safar, B.: Exploiting wordnet as background knowledge. In: The ISWC, vol. 7
17. Rosen-Zvi, M., Griffiths, T., Steyvers, M., Smyth, P.: The author-topic model for authors and documents. In: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, UAI 2004, pp. 487–494. AUAI Press, Arlington (2004)
18. Sanchez, E., Yamanoi, T.: Fuzzy Ontologies for the Semantic Web, pp. 691–699 (2006)
19. Stuckenschmidt, H.: Approximate information filtering on the semantic web. In: Jarke, M., Koehler, J., Lakemeyer, G. (eds.) KI 2002. LNCS (LNAI), vol. 2479, pp. 114–228. Springer, Heidelberg (2002)
20. Todorov, K., Geibel, P., Kühnberger, K.-U.: Mining concept similarities for heterogeneous ontologies. In: Perner, P. (ed.) ICDM 2010. LNCS, vol. 6171, pp. 86–100. Springer, Heidelberg (2010)
21. Xu, B., Kang, D., Lu, J., Li, Y., Jiang, J.: Mapping fuzzy concepts between fuzzy ontologies. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) KES 2005. LNCS (LNAI), vol. 3683, pp. 199–205. Springer, Heidelberg (2005)
22. Zadeh, L.A.: Fuzzy sets. *Information and Control* 8(3), 338–353 (1965)

Agent-Based Semantic Composition of Web Services Using Distributed Description Logics

Mourad Ouziri and Damien Pellier

LIPADE, Université Paris Descartes

45 rue des Saints Pères, 75006 Paris, France

{mourad.ouziri,damien.pellier}@parisdescartes.fr

Abstract. An important research challenge consists in composing web services in an automatic and distributed manner on a large scale. Indeed, most queries can not be satisfiable by one service and must be processed by composing several services. Each web service is often written by different designers and is described using the terms of their own ontology. Therefore, the composition process needs to deal with a variety of heterogeneous ontologies. In order to tackle this challenge, we propose an approach using Distributed Description Logics (DDL) to achieve the semantic composition of web services. DDL allows one to make semantic connections between ontologies and thus web services, as well as to reason to get a semantic composition of web services.

1 Introduction

The advent of Web services is an inevitable consequence of Web technology and its dissemination on a large scale, poses the problem of their automatic composition. The interoperability of Web services is guaranteed by three key XML-based standards. These standards have been defined to develop and deploy Web services: (1) SOAP (*Simple Object Access Protocol*) defines a communication protocol for Web services; (2) UDDI (*Universal Description Discovery and Integration*) is a registry service allowing the discovery of Web services and (3) WSDL (*Web Services Description Language*) is a language used to describe Web services which provides concepts to describe Web services from a syntactic point of view. Unfortunately, composing Web services requires more than the description of each service. In particular, it must be able to understand the other services and to learn how to interact with them. Thus, the lack of semantic tags in WSDL restricts their interoperability.

The concept of ontology is the key to improve Web services with semantics and interoperability. Ontologies enrich Web services with expressive and computer interpretable languages. They capture the semantics of Web services based on a formal representation of a set of concepts within a domain and the connections between those concepts and them, may be used to reason and compose Web services. Integrating ontologies into Web services could not only enhance the quality and the robustness of service discovery and invocation, but also pave the way for automated composition and seamless interoperation. Unfortunately, guaranteeing the interoperability and the automatic composition of Web services is not enough. This approach assumes that all the concepts are based on the same ontology. In practice, designers of Web services use their own ontologies

to describe their services. Therefore, we have to deal with the heterogeneous ontologies. For instance, how can one connect the terms “trip” and “journey” and indicate that they refer to the same concept? Dealing with a variety of different ontology-based descriptions of web services is still an open challenge.

In order to remove this obstacle, we propose a new approach based on distributed description logics. Distributed description logics is used to establish semantic connections between heterogeneous Web services. This approach has two main advantages:

1. To increase the interoperability between Web services by composing heterogeneous Web services. Our approach makes semantic composition of heterogeneous Web services. Even if the Web services are described using different and heterogeneous ontologies, our approach connects these ontologies using semantic connections between the terms of the ontologies. Then, we can use these connections to infer composable Web services automatically.
2. To reduce the complexity of the composition process by limiting it to only composable Web services. Indeed, traditional composition processes use planning techniques to compose Web services. The complexity of the composition process is limited by the number of services to be composed. This approach allows one to consider semantically composable services only as oppose to all available services.

The rest of the paper is organized as follows: section 2 proposes a synthesis of the related work, section 3 proposes a primary example, section 4 presents an overview of the distributed logic description and finally section 5 introduces our contribution.

2 Related Works

Over the previous decade, Web services have been the focus of a lot of research. The published literature concerns automatic discovery [18] and composition of web services [3]. Many approaches [12] [15] and languages [16], e.g., XLANG (*XML Business Process Language*), BPML (*Business Process Modeling Language*), WSFL (*Web Service Choreography Interface*), etc., were proposed to describe how web services can interact with each other with messages (taking to account the business logic and execution order of the interactions) and track the sequence of messages that may involve multiple parties and multiple sources (including customers, suppliers, and partners). In the rest of this paper, we are focused on the use of description logics [1] for web services discovering and composition:

Web services discovering: Matching is the process of searching the space of possible matches between supply and demand, finding the best available ones. Most of the works using description logics process for matching problems between a service provider and a service requester using standard satisfiability reasoning. Based on CLASSIC [7] structural subsumption algorithm, the best matches finding algorithm is proposed in [9]. The work proposed in [11] deals with the problems which occur in the matchmaking of incomplete service description because of the open-world assumption. In [5], proposed matchmaker architecture performs semantic matching of Web Services on the basis of input and output descriptions of semantic Web

Services. In [4] the service discovery is processed as a new instance of the problem of rewriting concepts using terminologies and calls the best covering problem. A hyper-graph-based algorithm to compute the best covers is proposed.

Web services composition: The web service composition problem consists in selecting a finite parallel or sequence of Web services to match a request. In [6], logical reasoning of description logics is used to perform e-Services composition. To do it, authors propose to re-express situation calculus action theories as a description logics knowledge base. In [13], description logics and AI planning are both used to compose services. This work does not deal with heterogeneous service descriptions. That is, the approach can not be composed if the services are described using multiple heterogeneous ontologies. Finally, the work presented in [19] uses description logics only to represent actions, plans and goals and to infer the subsumption connection between actions, plans and goals during plan generation, plan recognition, or plan evaluation. But, this work does not deal with service composition.

3 Description Logics Foundation

Description Logics (DL) [1] is a family of logics developed to represent complex hierarchical structures and to make reasoning facilities over these structures. A description logics knowledge base is composed of two parts: abstract knowledge (TBox) and concrete knowledge (ABox). Concrete knowledge ABox represents a set of facts, which are expressed by assertions on individuals of a real world. Abstract knowledge TBox is a set of concept and role descriptions. Concepts are unary predicates and roles are binary predicates. Semantics in DL is given by means of an interpretation function $I = (\Delta^I, .^I)$, where Δ^I is a set which represents the individuals of concrete knowledge and $.^I$ is an interpretation function defined as:

- $.^I(C) = C^I \subseteq \Delta^I$ for each concept C ;
- $.^I(R) = R^I \subseteq \Delta^I \times \Delta^I$ for each role R ;

Finally, a concept description is expressed using constructors (see [10] for examples).

Distributed description logics extend standard description logics to create descriptions that link concepts of multiple knowledge bases. Inspired by distributed first order logic [10], Distributed Description Logics (DDL) extends standard description logics as follows [8]:

1. Distributed ABox $DAB = (\{A_i\}_{i \in I}, \{r_{ij}\}_{i \neq j \in I})$: consists of a set of A-boxes and a set of individual correspondences $r_{ij} \subseteq \Delta_i \times \Delta_j$, where Δ_i and Δ_j are interpretation domains for A_i and A_j respectively.
2. Distributed TBox $DTB = (\{T_i\}_{i \in I}, \{B_{ij}\}_{i \neq j \in I})$: consists of a set of ordinary T-boxes and a set of so-called bridge rules, which express intentional assertions about connections. B_{ij} is a set of directional bridge rules from $KB_i(T_i, A_i)$ to $KB_j(T_j, A_j)$. A bridge rule that connects KB_i to KB_j is an axiom (in KB_j) of the following two forms:
 - Into-rules $i : C \sqsubseteq j : D$, i.e., in the knowledge base KB_j , the concept $j : D$ of KB_j subsumes the imported concept $i : C$ of KB_i . In the rest of the paper, we use the simple syntax $i : C \sqsubseteq j : D$ to express into-rules.

- Onto-rules $i : C \sqsupseteq j : D$, i.e., in the knowledge base KB_j , the concept $j : D$ of KB_j is subsumed by the imported concept $i : C$ of KB_i . In the rest of the paper, we use the simple syntax $i : C \sqsupseteq j : D$ to express onto-rules.
3. Distributed interpretation $DI = (\{I_i\}_{i \in I}, \{r_{ij}\}_{i \neq j \in I})$ consists of a set of ordinary interpretations of DTB T-Boxes and domain relations that interprets bridge rules as follows:
- Into-rule: $i : C \sqsubseteq j : D$, if $r_{ij}(C^{mj}) \subseteq D^{mj}$
 - Onto-rule: $i : C \sqsupseteq j : D$, if $r_{ij}(C^{mj}) \supseteq D^{mj}$

4 Agent-Based Semantic Composition of Web Services

Composing Web services requires the description of each service so that other services can understand its features. Unfortunately, semantic descriptions of services are not enough to allow automatic communication between services. That is, many terminologies can be used to describe services capabilities. Thus, we need to connect these terminologies to establish semantic and efficient communication between services. Our work focuses on semantic composition of services based on their functional aspects and no on their quality of services.

4.1 Primary Example

Let us consider an e-tourism application example where three agents A_1 , A_2 and A_3 provide hotel booking service in New-York and Washington, airplane transport service between France and the USA and restaurant service respectively. Suppose a person submits the query: "*I am in Paris and I would like to visit New-York for one week in July. I want to eat in a restaurant.*" The three agents A_1 , A_2 and A_3 must collaborate to process the query because none of them can solve the request alone. The communications between the agents to compose their services can be illustrated by the following informal dialogue:

- A1.1:** agent A_1 says: "*I can book a hotel from July 1st to July 7th. But someone else should propose a corresponding trip and restaurant with a complete menu*".
- A2.1:** agent A_2 says: "*I can offer a flight. But there is no flight available on July 1st. I can offer flights on July 3rd and July 9th*".
- A1.2:** agent A_1 says: "*OK, I can book a hotel from July 3rd to July 9th*".
- A3.1:** agent A_3 says: "*I can propose different restaurants with a full menu between July 3rd to July 9th*".

The three agents A_1 , A_2 and A_3 use different, incompatible terminologies to communicate. Thus, the above scenario of communication is not successful. That is, in A1.1, agent A_1 asks for a trip whereas in A2.1 agent A_2 offers a flight. Automatic agents do not make a semantic connection between the terms "trip" and "flight". When agent A_2 receives the request "*I need a trip*" from agent A_1 , it replies in A2.1 by "*I cannot offer trip*" (as it does not make the semantic connection between "trip" and "flight").

This small dialogue shows that the agents must be connected using semantic connections. We do so in two stages. First, we make a semantic description of the agents,

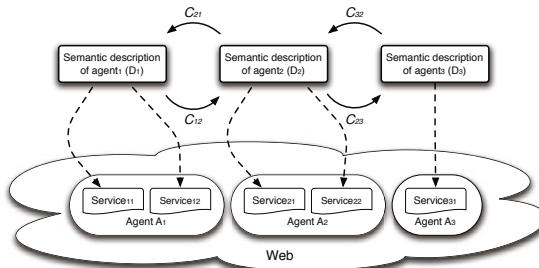


Fig. 1. Ontologies based annotation of Web services

especially of their provided services (each agent is described using a specific terminology). Secondly, we connect the agents between themselves using semantic connections between the agent descriptions. The proposed approach is shown in Fig. 1.

Services provided by agents are described using terms (concepts of ontology). For example, *Hotel*, *Trip*, *Flight*, *Date*, *Restaurant*, *NewYork*, *Menu*, *FullMenu*, etc. are some terms that can be used to describe agents A_1 , A_2 and A_3 of the above example. Relations C_{ij} between descriptions (see Fig 1) are semantic connections between the description D_i of the agent A_i and the description D_j of agent A_j from the point of view of A_j . These connections are directional and expressed from a particular agent's viewpoint.

The semantic connections C_{ij} connect terms of descriptions D_i to terms of D_j from the point of view of the agent A_j . This is done by importing terms of D_i in D_j using a set of assertions. In the above example, agents A_1 and A_2 can be connected using the assertion: $2 : Flight$ is a sub-concept of $1 : Trip$.

The service description in DL can be automatically generated from WSDL. However, the semantic connections are expressed manually in each peer then they are used by reasoning algorithms to discover automatically all the other implicit connections. We use only subsumption and disjunction relationships to connect concepts of different knowledge bases. Overlapping relationship is not considered because it may be expressed using subsumption and disjunction by refining concepts.

4.2 Service Composition Model

Given a set of Web services to compose, we propose the Web services composition model described as follows:

Definition 1 (Service description). *A service is described by the tuple $\langle D, P \rangle$ with D is a precise description of the task achieved by the service and P is a set of preconditions required by the service to achieve its task. Both elements are represented in a standard description logics.*

Definition 2 (Distributed directed knowledge base). *A distributed directed knowledge base $dKB < S_1, S_2, C_{12} >$ from service S_1 to service S_2 is defined by adding the following axioms : (i) Axioms that define service S_1 , (ii) Axioms that define service S_2 and (iii) Axioms C_{12} that connect the terms of service S_1 to those of service S_2 .*

Now, we define the concept of composable web services as follows:

Definition 3 (Service Composition Problem). *The service composition model is described by the tuple $\langle S, C \rangle$ where S is a set of services described and C is a set of semantic connections between these services. We use distributed description logics to represent this element.*

Given two services $S_i \langle D_i, P_i \rangle$ and $S_j \langle D_j, P_j \rangle$ in CS. The service S_i is composable with the service S_j , denoted by $S_i \circ S_j$, if the service S_j satisfies (subsumes) the preconditions P_i of S_j . That is, $dKB < S_i, S_j, C_{ij} > \models P_i \sqsubseteq D_j$.

4.3 Problem Formalization

Our formalization is based on description logics and their extensions to distributed description logics (see section 3). As shown in figure Fig. 1, the system is a collection of inter-related knowledge bases. Each agent is represented by description of provided services and semantic connections with the descriptions of the other agents. Formally, an agent is represented in a standard description logics TBox. Following the example of section 4.1, we describe services provided by agents A_1, A_2 and A_3 as follows:

Service S_1 of A_1 :

$Hotel \sqcap \exists location.NewYork \sqcap \exists arrival.Date \sqcap \exists departure.Date$

Preconditions: $\exists in.NewYork \sqcap (Trip \sqcap \exists hasDestination.NewYork \sqcap \leq 1 hasDestination) Restaurant \sqcap \exists hasMenu.Complete$

Effects: $HotelReservation$

Service S_2 of A_2 :

$Flight \sqcap \exists departure.Date \sqcap \exists departureAirport.FrenchAirport \sqcap \exists arrivalAirport.USAAirport$

Preconditions: $\exists oldLocation.France \sqcap \leq 1 oldLocation$

Effects: $\exists newLocation.France \sqcap \leq 1 newLocation$

Service S_3 of A_3 :

$Restaurant \sqcap \exists propose.Menu \sqcap \exists location.City \sqcap \leq 1 location$

Preconditions: $Restaurant \sqsubseteq \exists menuDate.Date \sqcap \exists menuType.(Light \sqcap Full \sqcap Vegetarian) \sqcap \leq 1 menuType$

Effects: $\exists toBeIn.NY \sqcap \leq 1 toBeIn$

The query: “I am in Paris and I will visit New-York in July. I want to eat a complete menu in a restaurant.” may be represented in description logics as:

- $Hotel \sqcap \exists location.NewYork \sqcap \exists arrival.July \sqcap \exists departure.July \quad (1 : q)$
- $\exists in.Paris \sqcap \leq 1 in \quad (1 : f)$

Using the subsumption reasoning of description logics, we have $1 : q \sqsubseteq S_1$ as $July \sqsubseteq Date$. Consequently, agent A_1 is able to execute the query. However, the preconditions of service S_1 implies that a trip is needed. Indeed, $\exists in.Paris \sqcap \leq 1 in (\exists in.NewYork \sqcap (Trip \sqcap \exists hasDestination.NewYork \sqcap \leq 1 hasDestination))$ is equivalent to $\exists in.Paris \sqcap \exists trip \sqcap \exists hasDestination.NewYork \sqcap \leq 1 hasDestination$.

$\exists \text{in}.NewYork \sqsubseteq 1 \text{in}$ or $\exists \text{in}.Paris \sqsubseteq 1 \text{in} \sqcap (\text{Trip} \sqcap \exists \text{hasDestination}.NewYork \sqsubseteq 1 \text{hasDestination})$. We have $\exists \text{in}.Paris \sqcap \exists \text{in}.NewYork \sqsubseteq 1 \text{in} \sqsubseteq \perp$ as $\text{Paris} \sqcap \text{NewYork} \sqsubseteq \perp$. Then, $\exists \text{in}.Paris \sqcap (\text{Trip} \sqcap \exists \text{hasDestination}.NewYork \sqsubseteq 1 \text{hasDestination})$ must be satisfied. This means that agent A_1 requires a trip. Agent A_1 must submit the description of the required trip to agents A_2 and A_3 . Agent A_2 should be able to satisfy the submitted requirement. However, as agents A_1 and A_2 use heterogeneous terminologies, *Trip* and *Flight* respectively, the reasoning services of description logics do not infer the connection between the requirement of A_1 and the offer of A_2 , expressed respectively by the descriptions: $\exists \text{in}.Paris \sqcap (\text{Trip} \sqcap \exists \text{hasDestination}.NewYork \sqsubseteq 1 \text{hasDestination})$ and $\text{Flight} \sqcap \exists \text{departure.Date} \sqcap \exists \text{departureAirport.FrenchAirport} \sqcap \exists \text{arrivalAirport.USAAirport}$ although terms *Trip* and *Flight*, *in* and *departureAirport*, *hasDestination* and *arrivalAirport* have the same meaning.

The solution we propose consists in connecting agent terminologies using DDL (Distributed Description Logics). Connecting agents A_i to A_j consists in making semantic connections between the preconditions of the connected agent A_i and description of the connecting agent A_j . For our example, we establish a semantic connection between agents A_1 and A_2 using the following distributed assertions added to the knowledge base of agent A_1 : (i) $A_1 : \text{Trip} \sqsupseteq A_2 : \text{Flight}$, (ii) $A_1 : \text{NewYork} \sqsubseteq A_2 : \text{USAAirport}$, (iii) $A_1 : \text{in} \sqsupseteq A_2 : \text{departureAirport}$ and (iv) $A_1 : \text{hasDestination} \sqsupseteq A_2 : \text{arrivalAirport}$.

4.4 Distributed Composition Algorithm

The distributed composition algorithm we propose is based on the distributed satisfiability reasoning proposed in [2]. It is based on *standard tableau algorithms* [2] and uses the message-based communication between local tableau algorithms. The distributed composition algorithm works at two levels: intra-agent level and inter-agent level. At the intra-agent level, the composition algorithm checks whether the agent supports the query. This is done using standard satisfiability reasoning (propagation rules) of description logics. If an agent supports the query, the algorithm verifies whether the facts of the query satisfy the preconditions of the agent. If the agent preconditions are satisfied, the algorithm ends. Otherwise, the algorithm follows at the inter-agents level to search agents that are able to satisfy the preconditions.

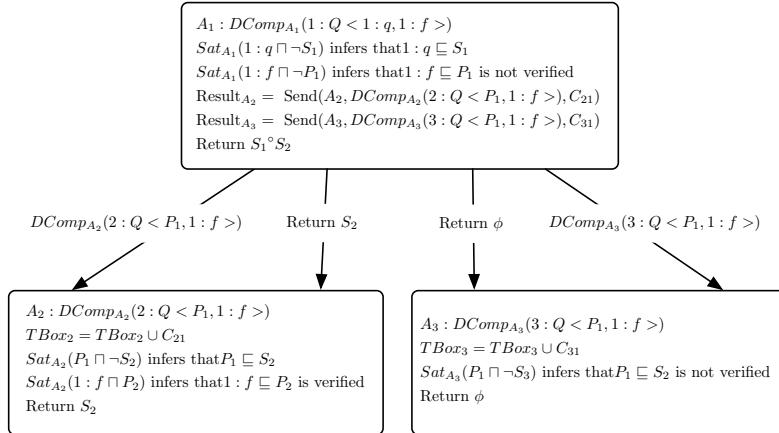
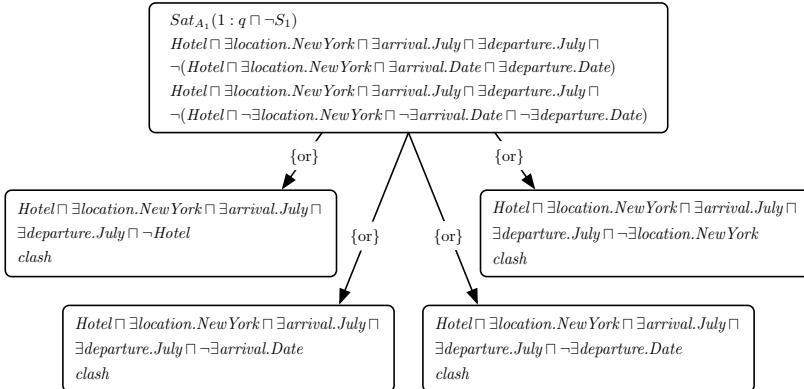
The proposed distributed reasoning algorithm, called $DComp_{A_i}(i : Q < i : q, i : f >)$ is based on the distributed satisfiability reasoning $DSat(C)$ proposed in [1]. The automatic composition works as follows:

In: a query $i : Q$ and initial fact $i : f$ expressed over agent A_i .

Out: set of composable services CS .

$DComp_{A_i}(i : Q < i : q, i : f >)$

1. Call $Sat_{A_i}(i : q \sqcap S_i)$ to check whether the query is supported by service S_i of agent A_i . This generates a constraint system (see figure 2), which is a set of assertions: $x : C$ and xRy where x and y are individuals, C is a concept description and R is a role description.
2. If $i : q \sqcap S_i \sqsubseteq \perp$, query is not supported by service S_i . Return $NULL$.
3. Call $Sat_{A_i}(P_i \sqcap \neg i : f)$ to check the subsumption $P_i \sqsubseteq i : f$:

**Fig. 2.** Beginning of the algorithm viewed by agent A_1 **Fig. 3.** Satisfiability reasoning and propagation rules

- (a) If $P_i \sqsubseteq i : f$, the preconditions of the query are satisfied by the facts given by the query. Service S_i does not require to be composed, it is able to process on its own query Q . The algorithm terminates and returns $CS = \{S_i\}$.
- (b) Otherwise, service S_i requires to be composed with services that are able to provide preconditions P_i .
- 4. For each agent A_j connected to A_i by connections C_{ji} , we verify whether A_j satisfies preconditions P_i of A_i :
 - (a) Add axioms of C_{ji} .
 - (b) Send $CS_j = DComp_{A_j}(j : Q < i : P_i, i : f >)$ to agent A_j with axioms of C_{ji} .
 - (c) If $CS_j \neq NULL$ then return $CS = \{S_i \circ CS_j\}$, else return $NULL$.

Let us illustrate this algorithm using the example of section 4.1. The query $1 : Q < 1 : q, 1 : f$ such that $1 : q = \text{Hotel} \sqcap \exists \text{location.NewYork} \sqcap \exists \text{arrival.July} \sqcap \exists \text{departure.July}$ and $1 : f = \exists \text{in.Paris}$ is expressed by agent A_1 . The algorithm starts with agent A_1 as shown in figure 2.

From figure 3, the query is submitted to agent A_1 , which applies standard satisfiability reasoning to decide whether the service provided by agent A_1 is able to process the query. The satisfiability reasoning is performed using propagation rules. Concept $1 : q \sqcap \neg S_1$ is satisfiable because all reasoning possibilities leads to *clash*. Then, query $1 : q$ is subsumed by service S_1 .

5 Conclusion

We propose in this paper a formal solution to compose heterogeneous web services. The proposed solution consists in describing services and preconditions provided by agents using description logics and making semantic connections between these descriptions. These inter-agents connections are formalized using distributed description logics. We propose a distributed reasoning algorithm that composes web services at a conceptual level with respect to agent connections. This algorithm uses the standard satisfiability algorithm of description logics. The use of distributed description logics allows to make more complete and consistent connections between the agents. That is, logical reasoning uses explicit connections to infer implicit ones since the number of agents to be connected in the semantic Web may be huge. Practicality, approaches based on logics and those based on planning are limited to few agents.

Evaluation of our approach is needed and important. Currently, our approach is under implementation using the JShop Planner [14]. Thus, we have only theoretical results. We are currently working on a complete evaluation of our approach, i.e., a theoretical and experimental analysis. The last one is difficult because no benchmarks exists. Therefore, we are currently developing our own set of benchmarks based on classical web services.

As future works, we plan to propose more reasoning facilities into one main direction. How to propose a complete model that integrates Web services composition at a conceptual level and practical composition at a planning level. Indeed, the Web services description used in our approach is very similar to the planning language such as PDDL (Planning Domain Description Language).

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.: *The Description Logic Handbook*. Cambridge University Press, Cambridge (2003)
2. Baader, F., Sattler, U.: An overview of tableau algorithms for description logics. *Studia Logica* 69(1), 5–40 (2001)
3. Benatallah, B., Dumas, M., Sheng, Q.: Facilitating the rapid development and scalable orchestration of composite web services. *Journal of Distributed and Parallel Databases* 17(1), 5–37 (2005)
4. Benatallah, B., Hacid, M., Léger, A., Rey, C., Toumani, F.: On automating web services discovery. *VLDB Journal* 14(1), 84–96 (2005)

5. Bener, A., Ozadalia, V., Ilhan, E.: Semantic matchmaker with precondition and effect matching using swrl. *Expert Systems with Applications* 36(5), 9371–9377 (2009)
6. Berardi, D., Calvanese, D., Giacomo, G.D., Lenzerini, M., Mecella, M.: e-service composition by description logics based reasoning. In: Proceedings of the International Workshop on Description Logics, pp. 75–84 (2003)
7. Borgida, A., Patel-Schneide, P.: A semantics and complete algorithm for subsumption in the classic description logic. *Journal of Artificial Intelligence Research* 1(1), 277–308 (1994)
8. Borgida, A., Serafini, L.: Distributed description logics: Assimilating information from peer sources. *Journal of Data Semantics* 1, 153–184 (2003)
9. Colucci, S., Sciascio, T.D., Domini, F., Mongiello, M.: Description logics approach to semantic matching of web services. *Journal of Computing and Information Technology* 11(3), 217–224 (2003)
10. Ghidini, C., Serafini, L.: Distributed first order logics. In: *Frontiers of Combining Systems 2. Studies in Logic and Computation*, pp. 121–140 (1998)
11. Grimm, S., Motik, B., Preist, C.: Matching semantic service descriptions with local closed-world reasoning. In: Sure, Y., Domingue, J. (eds.) *ESWC 2006. LNCS*, vol. 4011, pp. 575–589. Springer, Heidelberg (2006)
12. Levesque, H., Reiter, R., Lespérance, Y., Lin, F., Scherl, R.: GOLOG: a logic programming language for dynamic domains. *Journal of Logic Programming* 31(1-3), 59–84 (1997)
13. Lin, F., Qiu, L., Huang, H., Yu, Q., Shi, Z.: Description logic based composition of web services. In: Shi, Z.-Z., Sadananda, R. (eds.) *PRIMA 2006. LNCS (LNAI)*, vol. 4088, pp. 199–210. Springer, Heidelberg (2006)
14. Nau, D., Cao, Y., Lotem, A., Muoz-Avila, H., Muoz-Avila, H.: Shop: Simple hierarchical ordered planner. In: *IJCAI 1999*, pp. 968–973 (1999)
15. Peer, J.: A pop-based replanning agent for automatic web service composition. In: *Proceedings of the European Conference on Semantic Web*, pp. 47–61 (2005)
16. Peltz, C.: Web services orchestration - a review of emerging technologies, tools, and standards. Tech. rep., Hewlett Packard (2003)
17. Serafini, L., Tamlilin, A.: Local tableaux for reasoning in distributed description logics. In: *Description Logics* (2004)
18. Sycara, K., Paolucci, M., Ankolekar, A., Srinivasan, N.: Automated discovery, interaction and composition of semantic web services. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web* 1(1), 27–46 (2003)
19. Yolanda, G.: Description logics and planning. *AI Magazine* 26(2), 73–84 (2005)

Temporal Reasoning for Supporting Temporal Queries in OWL 2.0

Sotiris Batsakis, Kostas Stravoskoufou, and Euripides G.M. Petrakis

Department of Electronic and Computer Engineering
Technical University of Crete (TUC)
Chania, Greece

batsakis@softnet.tuc.gr, {kgstravo, petrakis}@intelligence.tuc.gr

Abstract. We propose an approach for reasoning and querying over temporal information in OWL ontologies. Representing both qualitative temporal (i.e., information whose temporal extents are unknown such as “before”, “after” for temporal relations) in addition to quantitative information (i.e., where temporal information is defined precisely e.g., using dates) is a distinctive feature of the proposed ontology. Qualitative representations are very common in natural language expressions such as in free text or speech and can be proven to be valuable in the Semantic Web. Reasoning rules applying over temporal relations, infer implied relations, detect inconsistencies and retain soundness, completeness and tractability over the supported sets of relations using path consistency. Temporal representations are defined on time instants rather than on intervals (as it is typical in the literature), resulting into simpler yet equivalent representations. A SPARQL-based temporal query language capable of exploiting the characteristics of the underlying representation is also implemented and discussed.

1 Introduction

Ontologies offer the means for representing high level concepts, their properties and their interrelationships. Dynamic ontologies will in addition enable representation of information evolving in time. Representation of dynamic features calls for mechanisms allowing for uniform representation of the notions of time (and of properties varying in time) within a single ontology. Methods for achieving this goal include (among others), temporal description logics [4], concrete domains [6], property labeling [5], versioning [2], named graphs [8], reification¹ and the 4D-fluents (perdurantist) approach [3].

Welty and Fikes [3] showed how quantitative temporal information (i.e., in the form of temporal intervals whose start and end points are defined) and the evolution of concepts in time can be represented effectively in OWL using the so called “4D-fluents approach”. In [1] this approach was extended in certain ways: (a) The 4D fluents mechanism was enhanced with qualitative (in addition to quantitative) temporal expressions allowing for the representation of temporal intervals with unknown starting and ending points by means of their relation (e.g., “before”, “after”) to other time intervals. SWRL and OWL 2.0 constructs (e.g., disjoint properties) are combined, offering a sound and complete reasoning procedure ensuring path consistency [13].

¹ <http://www.w3.org/TR/swbp-n-aryRelations/>

This approach still suffers from two disadvantages: (a) relying on an interval-based representation didn't allow for reasoning over time instants similarly to intervals, (b) Implementing path consistency in SWRL called for additional temporal relations thus complicating the representation. The proposed approach tackles both these problems. In addition, we propose an extension to SPARQL with temporal operators for effectively querying over temporal information in OWL ontologies. The query language is independent of the 4D fluent representation introduced into this work (i.e., may work equally well with any other representation). To the best of our knowledge, this is the first work dealing with both qualitative and quantitative temporal information in ontologies and at the same time handling all issues referred to above within the same representation.

Related work in the field of knowledge representation is discussed in Section 2. This includes issues related to representing and reasoning over information evolving in time. The proposed ontology is presented in Section 3 and the corresponding reasoning mechanism in Section 4. The temporal query language is presented in Section 5 followed by evaluation in Section 6 and conclusions and issues for future work in Section 7.

2 Background and Related Work

The OWL-Time temporal ontology² describes the temporal content of Web pages and the temporal properties of Web services. Apart from language constructs for the representation of time in ontologies, there is still a need for mechanisms for the representation of the evolution of concepts (e.g., events) in time.

Temporal Description Logics (TDLs) [4] extend standard description logics (DLs) that form the basis for semantic Web standards with additional constructs such as “always in the past”, “sometime in the future”. TDLs offer additional expressive capabilities over non temporal DLs but they require extending OWL syntax and semantics with the additional temporal constructs. Representing information concerning specific time instants requires support for concrete domains. *Concrete Domains* [6] introduce datatypes and operators based on an underlying domain (such as decimal numbers). The concrete domains approach requires introducing additional datatypes and operators to OWL, while our work relies on existing OWL constructs. This is a basic design decision in our work. TOWL [11] is an approach combining 4D fluents with concrete domains but didn't support qualitative relations, path consistency checking (as this work does) and is not compatible with existing OWL editing, querying and reasoning tools (e.g., Protege, Pellet, SPARQL).

Versioning [2] suggests that the ontology has different versions as time evolves. When a change takes place, a new version is created. Versioning suffers from several disadvantages: (a) changes even on single attributes require that a new version of the ontology be created leading to information redundancy, (b) searching for events requires exhaustive searches in multiple versions of the ontology, (c) it is not clear how the relation between evolving classes is represented. *Named Graphs* [8] represent the temporal context of a property by inclusion of a triple representing the property in a named graph (i.e., a subgraph into the RDF graph of the ontology specified by a distinct name). The

² <http://www.w3.org/TR/owl-time/>

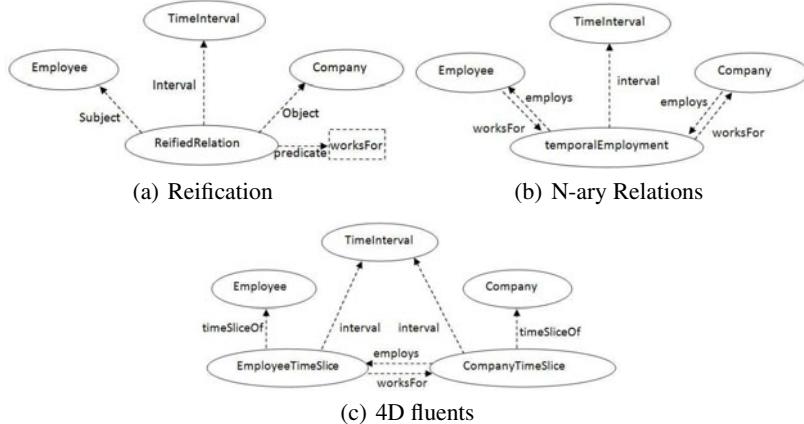


Fig. 1. Example of (a) Reification (b) N-ary Relations and (c) 4D-fluents

default (i.e., main) RDF graph contains definitions of interval start and end points for each named graph, so that a temporal property is represented by the start and end points corresponding to the temporal interval that the property holds. Named graphs are neither part of the OWL specification³ (i.e., there are not OWL constructs translated into named graphs) nor they are supported by OWL reasoners.

Reification is a general purpose technique for representing *n*-ary relations using a language such as OWL that permits only binary relations. Specifically, an *n*-ary relation is represented as a new object that has all the arguments of the *n*-ary relation as objects of properties. For example, if the relation *R* holds between objects *A* and *B* at time *t*, this is expressed as *R(A,B,t)*. In OWL this is expressed as a new object with *R*, *A*, *B* and *t* being objects of properties. Fig. 1(a) illustrates the relation *WorksFor(Employee, Company, TimeInterval)* representing the fact that an employee works for a company during a time interval. The extra class “ReifiedRelation” is created having all the attributes of the relation as objects of properties. Reification suffers mainly from two disadvantages: (a) a new object is created whenever a temporal relation has to be represented (this problem is common to all approaches based on OWL) and (b) offers limited OWL reasoning capabilities [3]. Because relation *R* is represented as the object of a property, OWL semantics over properties (e.g., inverse properties) are no longer applicable (i.e., the properties of a relation are no longer associated directly with the relation itself). CNTRO [4] describes a temporal medical ontology using reification.

The N-ary relations approach suggests representing an *n*-ary relation as two properties each related with a new object (rather than as the object of a property, as reification does). This approach requires only one additional object for every temporal relation, maintains property semantics but (compared to the 4D-fluents approach discussed in this work) suffers from data redundancy in the case of inverse and symmetric properties (e.g., the inverse of a relation is added explicitly twice rather than once as in the 4D fluents approach). This is illustrated in Fig 1(b). Furthermore, domains and ranges of

³ <http://www.w3.org/TR/owl2-syntax/>

properties have to be adjusted taking into account the classes of intermediate objects representing the relation (for example the *worksFor* relation is no longer a relation having as object an individual of class *Company* and subject of class *Employee* as they are now related to the new object *TemporalEmployment*).

The *4D-fluent* (perdurantist) approach [3] shows how temporal information and the evolution of temporal concepts can be represented in OWL. Concepts evolving in time are represented as 4-dimensional objects with the 4th dimension being the time (*timeslices*). Time instances and time intervals are represented as instances of a *TimeInterval* class, which in turn is related with concepts varying in time as shown in Fig. I(c). Changes occur on the properties of the temporal part of the ontology keeping the entities of the static part unchanged. The 4D-fluent approach still suffers from proliferation of objects since it introduces two additional objects for each temporal relation (instead of one in the case of reification and N-ary relations). The N-ary relations approach referred to above is considered to be an alternative to the 4D fluents approach considered into this work.

3 Temporal Ontology

Following the approach by Welty and Fikes [3], to add the time dimension to an ontology, classes *TimeSlice* and *TimeInterval* with properties *TimeSliceOf* and *TimeInterval* are introduced. Class *TimeSlice* is the domain class for entities representing temporal parts (i.e., “time slices”) and class *TimeInterval* is the domain class of time intervals. A time interval holds the temporal information of a time slice. Property *TimeSliceOf* connects an instance of class *TimeSlice* with an entity, and property *interval* connects an instance of class *TimeSlice* with an instance of class *TimeInterval*. Properties having a temporal dimension are called *fluent properties* and connect instances of class *TimeSlice* (as in Fig. I(c)).

In our previous work [1] the 4D-fluents representation was enhanced with qualitative temporal relations (i.e., relations holding between time intervals whose starting and ending points are not specified) by introducing temporal relationships as object relations between time intervals. A temporal relation can be one of the 13 pairwise disjoint Allen’s relations [7] of Fig. 2. Notice that, temporal instants still cannot be expressed; subsequently, relations between time instants or between instants and intervals cannot be expressed explicitly.

In this work, an instant-based (or point-based) approach is adopted. Also, definitions for temporal entities (e.g., instants and intervals) are provided by incorporating OWL-Time into the same ontology. Each interval (which is an individual of the *ProperInterval* class) is related with two instants (individuals of the *Instant* class) that specify its starting and ending points using the *hasBeginning* and *hasEnd* object properties respectively. In turn, each *Instant* can be related with a specific date represented using the concrete *dateTime* datatype. One of the *before*, *after* or *equals* relations may hold between any two temporal instants with the obvious interpretation. In fact, only relation *before* is needed since relation *after* is defined as the inverse of *before* and relation *equals* can be represented using the *sameAs* OWL keyword applied on temporal instants. In this work, for readability we use all three relations. Notice

also that, property *before* may be also qualitative when holding between time instants or intervals whose values or end points are not specified. This way, we can assert and infer facts beyond the ones allowed when only instants or intervals with known values (e.g., dates) or end-points are allowed.

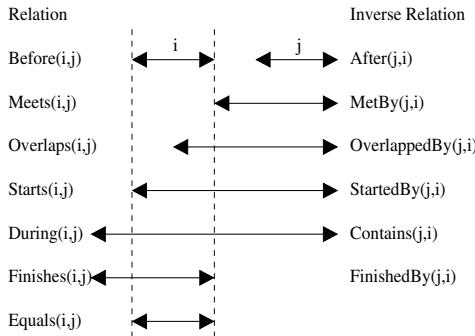


Fig. 2. Allen's Temporal Relations

Relations between intervals are expressed as relations between their starting and ending points which in turn are expressed as a function of the three possible relations between points (time instants) namely *equals*, *before* and *after* denoted by “ $=$ ”, “ $<$ ” and “ $>$ ” respectively, forming the so called “point algebra” [2]. Let $i_1 = [s_1, e_1]$ and $i_2 = [s_2, e_2]$ be two intervals with starting and ending points s_1, s_2 and e_1, e_2 respectively; then, the 13 Allen relations of Fig. 2 are rewritten as follows:

$$\begin{aligned}
 i_1 \text{ before } i_2 &\equiv e_1 < s_2 \\
 i_1 \text{ equals } i_2 &\equiv s_1 = s_2 \wedge e_1 = e_2 \\
 i_1 \text{ overlaps } i_2 &\equiv s_1 < s_2 \wedge e_1 < e_2 \wedge e_1 > s_2 \\
 i_1 \text{ meets } i_2 &\equiv e_1 = s_2 \\
 i_1 \text{ during } i_2 &\equiv s_1 > s_2 \wedge e_1 < e_2 \\
 i_1 \text{ starts } i_2 &\equiv s_1 = s_2 \wedge e_1 < e_2 \\
 i_1 \text{ finishes } i_2 &\equiv s_1 > s_2 \wedge e_1 = e_2
 \end{aligned}$$

The relations *after*, *overlappedby*, *metby*, *contains*, *startedby* and *finishedby* are the inverse of *before*, *overlaps*, *meets*, *during*, *starts* and *finishes* and are defined accordingly (by interchanging s_1, s_2 and e_1, e_2 in their respective definitions). These temporal relations and the corresponding reasoning mechanism are integrated both in the 4D-fluents and the n-ary based ontologies.

4 Temporal Reasoning

Reasoning is realized by introducing a set of SWRL⁴ rules operating on temporal relations. Reasoners that support DL-safe rules such as Pellet⁵ can be used for inference and consistency checking over temporal relations. Alternatively, OWL axioms on temporal properties can be used instead of SWRL. As we shall see later, this approach cannot guarantee decidability and is therefore not compatible with W3C specifications. The three temporal relations between points are *before*, *after* and *equals*, denoted by symbols “<”, “>”, “=” respectively. Table II illustrates the set of reasoning rules based on the composition of existing relation pairs.

Table 1. Composition Table for point-based temporal relations

Relations	<	=	>
<	<	<, <, =, >	
=	<	=	>
>	<, =, >	>	>

The composition table represents the result of the composition of two temporal relations. For example, if relation R_1 holds between $instant_1$ and $instant_2$, and relation R_2 holds between $instant_2$ and $instant_3$, then the entry of the Table II corresponding to line R_1 and column R_2 denotes the possible relation(s) holding between $instant_1$ and $instant_3$. Also, the three temporal relations are declared as pairwise disjoint, since they can't simultaneously hold between two instants. Not all compositions yield a unique relation as a result. For example, the composition of relations *before* and *after* yields all possible relations as a result. Because such compositions doesn't yield new information these rules are discarded. Rules corresponding to compositions of relations R_1 , R_2 yielding a unique relation R_3 as a result are retained (7 out of the 9 entries of Table II are retained) and are expressed in SWRL using rules of the form:

$$R_1(x, y) \wedge R_2(y, z) \rightarrow R_3(x, z)$$

The following is an example of such a temporal inference rule:

$$before(x, y) \wedge equals(y, z) \rightarrow before(x, z)$$

A series of compositions of relations may imply relations which are inconsistent with existing ones (for example the rule referred to above will yield a contradiction if $after(x, z)$ has been asserted into the ontology for specific values of x, y, z). Consistency checking is achieved by ensuring path consistency [13]. Path consistency is implemented by consecutively applying the following formula:

$$\forall x, y, k \ R_s(x, y) \leftarrow R_i(x, y) \cap (R_j(x, k) \circ R_k(k, y))$$

⁴ <http://www.w3.org/Submission/SWRL/>

⁵ <http://clarkparsia.com/pellet/>

representing intersection of compositions of relations with existing relations (symbol \cap denotes intersection, symbol \circ denotes composition and R_i, R_j, R_k, R_s denote temporal relations). The formula is applied until a fixed point is reached (i.e., the consecutive application of the rules above doesn't yield new inferences) or until the empty set is reached, implying that the ontology is inconsistent. Thus, in addition to rules implementing compositions of temporal relations, a set of rules defining the result of intersecting relations holding between two instances must also be defined in order to implement path consistency. These rules are of the form:

$$R_1(x, y) \wedge R_2(x, y) \rightarrow R_3(x, y)$$

where R_3 can be the empty relation. For example, the intersection of the relation representing the disjunction of *before*, *after* and *equals* (abbreviated as *ALL*), and the relation *before* yields the relation *before* as result:

$$ALL(x, y) \wedge before(x, y) \rightarrow before(x, y)$$

The intersection of relations *before* and *after* yields the empty relation, and an inconsistency is detected:

$$before(x, y) \wedge after(x, y) \rightarrow \perp$$

As shown in Table 1, compositions of relations can yield one of the following four relations: *before*, *after*, *equals* and the disjunction of these three relations. Intersecting the disjunction of all three relations with any of these leaves existing relations unchanged. Intersecting any one of the tree basic (non disjunctive) relations with itself also leaves relations unaffected. Only compositions of pairs of different basic relations affect the ontology by yielding the empty relation as a result, thus detecting an inconsistency. By declaring the three basic relations (*before*, *after*, *equals*) as pairwise disjoint, all intersections that can affect the ontology are defined. Path consistency is implemented by defining compositions of relations using SWRL rules and declaring the three basic relations as disjoint. Notice that, path consistency is sound and complete when applied on the three basic relations [13].

Alternatively, we can define the composition of *before* with itself as a transitivity axiom rather than by an SWRL rule. In this case, there would be no need for SWRL rules applying only on named individuals into the ontology ABox. The resulting representation will apply on the TBox as well. However, there is an obstacle in this approach forcing the use of SWRL rules: the relation *before* must be declared as transitive in order to infer implied relations and disjoint with *after*, its inverse relation, (also *before* is asymmetric and irreflexive) in order to detect inconsistencies. But OWL specifications⁶ disallow the combination of transitivity and disjointness (or asymmetry) axioms on a property because they can lead to undecidability [9]. This restriction is necessary in order to guarantee decidability in reasoning with OWL 2 DL.

⁶ http://www.w3.org/TR/2009/REC-owl2-syntax-20091027/#The_Restrictions_on_the_Axiom_Closure

5 Temporal Query Language

We also design and implement a SPARQL-like temporal query language based on the idea of introducing a set of temporal operators (i.e., the AT and Allen operators) on top of SPARQL. Query execution relies on the intermediate translation of the temporal operators into an equivalent SPARQL query. This translation depends on the underlying ontology representation (e.g., it is expressed as a set operations over fluent properties in case the 4D-fluents representation is used). In this work, we have experimented with both representations described in Section 2 namely, N-ary relations and the 4D-fluent approach. The motivation of extending SPARQL with new operators (rather than using SPARQL with its existing operators for querying temporal information) is that this approach offers additional flexibility in expressing temporal queries concisely, while ensuring independence of the temporal expressions from the peculiarities of the underlying ontological representation (i.e., query syntax is the same regardless of the temporal representation adopted). The query language inherits SPARQL syntax and semantics (e.g., queries over non-temporal information is expressed in SPARQL) with the addition of the temporal operators.

We introduce clauses AT, SOMETIME_AT and ALWAYS_AT for comparing a fluent property (i.e., the time interval during which the property holds true) with a time period (time interval) or time point. Such queries return fluents holding true at the specified time interval or point. Queries involving static properties (properties not changing in time) are issued as normal SPARQL queries applied on the static part of the ontology. Operator SOMETIME_AT returns fluents holding for intervals that share common time points with the interval in question. Operator ALWAYS_AT returns fluents holding for intervals which contain all points of the interval in question. For example the following query retrieves the name of the company “x” where employee “y” was always working for, during the specified interval:

```
select ?x ?y where {
?x ex1:hasEmployee ?y ALWAYS_AT
“2007 – 02 – 01T00 : 00 : 00Z”, “2007 – 02 – 05T00 : 00 : 00”}
```

The “AT” operator returns fluents holding at intervals that contain the time point in question. Notice that, integrating interval and instance representations allows for inferring relations between points and intervals in addition to relations between intervals. For example the following query retrieves the name of the company “x” that employee “y” was working for, at the specified date:

```
select ?x ?y where {
?x ex1:hasEmployee ?y AT “2007 – 02 – 05T00 : 00 : 00”}
```

The following Allen operators are also supported: BEFORE, AFTER, MEETS, METBY, OVERLAPS, OVERLAPPEDBY, DURING, CONTAINS, STARTS, STARTEDBY, ENDS, ENDEDBY and EQUALS, representing the relations holding between two time intervals specified (operators BEFORE and AFTER support temporal points as well). In this work, relations can be quantitative (i.e., involving specific temporal instants or intervals) or qualitative (i.e., the exact values of temporal instants or intervals

are unknown or not specified). For example the following query retrieves the name of the company “x” that employee “y” was working for, before *Employee2* worked for *Company1*:

```
select ?x ?y where {
?x ex1:hasEmployee ?y before
ex1:Company1 ex1:hasEmployee ex1:Employee2}
```

6 Evaluation

Reasoning is achieved by employing DL-safe rules expressed in SWRL that apply on named individuals in the ontology A-box, thus retaining decidability while offering a sound and complete inference procedure for asserted temporal intervals or instants. Furthermore, reasoning has polynomial time complexity since only the basic Allen or point-based relations are supported [12][13].

Because any time interval can be related with every other interval with one basic Allen relation (basic Allen relations are mutually exclusive) between n intervals, at most $(n - 1)^2$ relations can be asserted and this also holds in case of instants. Furthermore, path consistency has $O(n^5)$ worst time complexity (with n being the number of intervals or instants) and is sound and complete for the supported sets of relations. In the most general case where disjunctive relations are supported in addition to the basic ones, any time interval (or time instant) can be related with every other interval (or instant) by at most k relations, where k is the size of the set of supported relations. Therefore, for n intervals or instants, using $O(k^2)$ rules, at most $O(kn^2)$ relations can be asserted into the knowledge base. In the case of temporal instants [13], qualitative relations on time instants form a tractable set (i.e., a set of relations applying path consistency on this is a sound and complete method) if the relation \neq (i.e., a temporal instant is before *or* after another instant) is excluded. Reasoning can be extended with disjunctive relations such as \geq denoting that an instant is *after or equals* to another.

In this work, a point-based representation is adopted for handling both time instants and intervals. Relations between intervals are expressed as a function of relations between their end-points. A representation relying on intervals is also possible. However, since the number of basic relations is 13 (Fig. 2) and because all possible disjunctions appearing in the supported tractable set must also be supported, the representation may become particularly involved. Notice also that, time instants, the same as semi-closed temporal intervals (i.e., intervals with only one quantitative defined end-point) cannot be represented efficiently in an interval-based representation which doesn’t handle points. For example, if interval *A* *contains* interval *B* and point *C* is *into* interval *B* we can infer using the point based representation that *C* is *into* interval *A*.

7 Conclusions and Future Work

The contributions of this work are twofold: First, we propose an approach for handling temporal knowledge in ontologies using OWL and SWRL. It handles both time instants and temporal intervals (and also semi-closed intervals) equally well using a sound

and complete inference procedure based on path consistency. Second, we introduce a SPARQL-based temporal query language for expressing temporal queries. Extending our approach for spatial and spatio-temporal information and addressing scalability issues are directions of future work.

References

1. Batsakis, S., Petrakis, E.G.M.: SOWL: Spatio-temporal Representation, Reasoning and Querying over the Semantic Web. In: 6th International Conference on Semantic Systems, Graz, Austria (September 1-3, 2010)
2. Klein, M., Fensel, D.: Ontology Versioning for the Semantic Web. In: International Semantic Web Working Symposium (SWWS 2001), California, USA, pp. 75–92 (July-August 2001)
3. Welty, C., Fikes, R.: A Reusable Ontology for Fluents in OWL. *Frontiers in Artificial Intelligence and Applications* 150, 226–236 (2006)
4. Artale, A., Franconi, E.: A Survey of Temporal Extensions of Description Logics. *Annals of Mathematics and Artificial Intelligence* 30(1-4) (2001)
5. Gutierrez, C., Hurtado, C., Vaisman, A.: Introducing Time into RDF. *IEEE Trans. on Knowledge and Data Engineering* 19(2), 207–218 (2007)
6. Lutz, C.: Description logics with concrete domains - A survey. In: *Advances in Modal Logics*, King's College, vol. 4 (2003)
7. Allen, J.F.: Maintaining Knowledge About Temporal Intervals. *Communications of the ACM* 26, 832–843 (1983)
8. Tappolet, J., Bernstein, A.: Applied Temporal RDF: Efficient Temporal Querying of RDF Data with SPARQL. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) *ESWC 2009. LNCS*, vol. 5554, pp. 308–322. Springer, Heidelberg (2009)
9. Horrocks, I., Kutz, O., Sattler, U.: The Even More Irresistible SROIQ. In: Proc. KR 2006, Lake District, UK (2006)
10. Grau, B.C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., Sattler, U.: OWL 2: The Next Step for OWL. In: *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 6, pp. 309–322 (2008)
11. Milea, V., Frasincar, F., Kaymak, U.: Knowledge Engineering in a Temporal Semantic Web Context. In: The Eighth International Conference on Web Engineering, ICWE 2008 (2008)
12. Nebel, B., Burckert, H.J.: Reasoning about Temporal Relations: A Maximal Tractable Sub-class of Allen's Interval Algebra. *Journal of the ACM (JACM)* 42(1), 43–66 (1995)
13. van Beek, P., Cohen, R.: Exact and approximate reasoning about temporal relations. *Computational Intelligence* 6(3), 132–147 (1990)
14. Tao, C., Wei, W.Q., Solbrig, H.R., Savova, G., Chute, C.G.: CNTRQ: A Semantic Web Ontology for Temporal Relation Inferencing in Clinical Narratives. In: AMIA Annual Symp. Proc. 2010, pp. 787–791 (2010)

An Algorithm for Finding Gene Signatures Supervised by Survival Time Data

Stefano M. Pagnotta¹ and Michele Ceccarelli^{1,2}

¹ Department of Science

University of Sannio, 82100, Benevento, Italy

² Bioinformatics CORE, BIOGEM s.c.a.r.l., Contrada Camporeale,
Ariano Irpino, Italy
ceccarelli@unisannio.it, pagnotta@unisannio.it

Abstract. Signature learning from gene expression consists into selecting a subset of molecular markers which best correlate with prognosis. It can be cast as a feature selection problem. Here we use as optimality criterion the separation between survival curves of clusters induced by the selected features. We address some important problems in this fields such as developing an unbiased search procedure and significance analysis of a set of generated signatures. We apply the proposed procedure to the selection of gene signatures for Non Small Lung Cancer prognosis by using a real data-set.

1 Introduction

Gene expression profiling allows to characterize, analyze and classify tissues as function of the amount of transcript present in the cell for each of tens of thousand of genes. Many complex diseases such as cancer, can be studied by observing the variability profile of expression for thousands of genes by using microarray technologies [19]. Clustering of global gene expression patterns has been used to identify, at a molecular level, distinct subtypes of cancer, distinguished by extensive differences in gene expression, in diseases that were considered homogeneous based on classical diagnostic methods [13][15][18]. The molecular subtypes can be then associated with different clinical outcomes.

One of important open question in this area is related to the methods to extract suitable features from global gene expression that best correlate with clinical behavior to create prognostic signatures [7]. For example in breast cancer, a poor prognosis gene expression signature in the primary tumor can accurately predict the risk of subsequent metastases, independent of other well known clinico-pathologic risk factors [8]. Most of these approaches are based on expert knowledge to select, from thousands of genes, molecular markers which can be associated to prognosis [7]. Just recently some new methods grounded on the data mining and machine learning theory have appeared [3]. “Signature learning” is a rather new research topic in Bioinformatics, however it can be modeled as a standard problem of feature selection [11]: given a set of expression levels of N genes (of order of tens of thousands) in M different conditions or samples (of

order of hundres) and a target classification C (prognostic classes), the problem consists in selecting a subspace of n features from the N dimensional observation that “optimally” characterizes C . Given the optimality criterion, a search algorithm must be devised to find the best subspace from the 2^N possibilities. This poses two important questions: (i) how to select the optimality criterion; (ii) since exhaustive search is not viable, how to devise a suitable heuristic search. For the first question we adopt, as in [3], the survival time measured by using the log-rank test [16] between the separation in group induced by the selected features. For the second problem we propose a novel procedure which integrates several signatures generated by a simple greedy algorithm.

The paper is organized as follows, the next section reports a description of the signature learning algorithms, whereas in the section of Results we report an evaluation of the method on a real dataset

2 An Unbiased Procedure for Signature Learning and Discriminative Gene Ranking

Here we present a novel algorithm for signature learning from gene expression data, we extend the method proposed in [3].

2.1 The mSD Algorithmn

Boutros et al. [3] introduced a promising algorithm, called modified Steepest Descent (mSD), to discover prognostic gene signatures related to the non-small-cell lung cancer (NSCLC). The data necessary for the algorithm are an $N \times M$ matrix of expression levels of a set of N genes measured on M samples; and, for each sample, the survival time t_i , $i = 1, 2, \dots, M$ to death (eventually right censored if the corresponding person exits from the clinical study).

The expression levels are used to cluster the samples into two groups. The times are generally used to estimate the survival function $S(t)$ that measures the fraction of people which survive after the time t . Actually two survival curves $S_1(t)$ and $S_2(t)$ are considered, one for each of the two groups provided by the clustering. When $S_1(t) > S_2(t)$, for all $t > 0$, then the samples used to estimate $S_1(t)$ define the *good prognosis group*, while those used for $S_2(t)$ define the *poor prognosis group*. At present the survival curves are not estimated but their difference is inferentially measured through the p -value of the log-rank test [16] concerning the null hypothesis of no difference is in the population survival curves for the good and poor prognosis group. The p -value is used as a measure of similarity between the population survival curves so that they are as different as the p -value is close to 0. The basic idea of the signature finding algorithm was to find the sequence of genes having the properties (1) to separate the samples in two groups according to an unsupervised strategy, then (2) the estimated population survival curves related to the good and poor prognosis groups have to be maximally different. To this aim, the first step is to select a single gene having the properties (1) and (2); then the set of genes is scanned again to find

the best pair of genes still having the properties (1) and (2) where one element of the pair is that found in the previous step. The scanning is iterated until a sequence of $L > 0$ genes is found so that no sequence of $L+1$ elements can reach a separation of the survival curves better than the sequence of K genes.

2.2 A Heuristic Search for the Signature

The algorithms described above has been applied to find a new signature for NSLC. However it has some drawbacks: (i) it is a greedy procedure and the genes of the signature are selected one at time in a sequential order, but the results are strongly influenced by the order in which gene are added to the signature, in particular, starting from a different gene can lead to a completely different signature; this means that there are many local minima and the deterministic behavior of mSD can be easily trapped in a local minimum, for this reason our algorithm tries to generate several signature from different genes chosen on the basis of a statistical test; (ii) as reported in the Results section, the genes of the signature can be used for diagnostic purposes if they have a different behavior, with a given statistical significance, in the two populations (poor or good prognosis), we observed that not all genes in the signature generated by the mSD algorithm have this property, therefore we develop a pruning strategy (iii) finally the mSD can suffer by the selection bias, we developed a feature selection procedure using an external cross validation scheme.

Our algorithm consists of four separate steps: (1) find candidate seed genes; (2) generate a signature for each seed gene; (3) prune the signatures by statistical inference; (4) integrate signatures by gene ranking.

Finding the seed genes. The first step of our algorithm is aimed at selecting a set of genes which can be a reliable starting point to expand the signature. We use as starting seed of the signature a set of genes according to two main conditions: (i) a bimodal distribution of the expression levels, (ii) ability to separate the dataset in two groups on the basis of the survival analysis test. In particular, for each gene i consider the sequence $\mathbf{g}_i = \{x_{i,j}\}_{j=1,\dots,M}$, where $x_{i,j}$ is the expression level of gene i in the sample j . The sequence \mathbf{g}_i is clustered in two subsets S_1 and S_2 by using a k -median algorithm ($k = 2$). The bimodality hypothesis is checked by using the Bayesian Information Criterion (BIC) [20]. In particular, the whole sequence \mathbf{g}_i is checked against a normal distribution:

$$\text{BIC}_1 = -2 \sum_{j=1}^M \log \phi(x_{i,j}; \mu, \sigma^2) + 2 \log M$$

and a mixture of two gaussians (with means and variances computed from S_1 and S_2),

$$\text{BIC}_2 = -2 \sum_{i=1}^n \log f_{\text{mix}}(x_{i,j}; \mu_1, \sigma_1^2, \mu_2, \sigma_2^2, \tau) + 5 \log M$$

where: $f_{\text{mix}}(x; \mu_1, \sigma_1^2, \mu_2, \sigma_2^2, \tau) = \tau \phi(x, \mu_1, \sigma_1^2) + (1 - \tau) \phi(x, \mu_2, \sigma_2^2)$.

The set of seed genes is selected by considering those genes with $\text{BIC}_2 < \text{BIC}_1$ and the p -value of the log-rank test between $S_1(t)$ and $S_2(t)$ less than a specified threshold α (in all the experiments reported below we use $\alpha = 0.05$).

Generation of gene signatures. We assume that a set of $h \ll m$ candidate starting genes j_1, j_2, \dots, j_h has been found from the previous step. For each of these genes we generate a g signature having one of these gene as the starting point to expand the signature. Given a partial sequence of l genes having a log-rank test p_l -value, it seems from the mSD algorithm that the searching of the signature is performed by selecting the $l + 1$ gene such that $p_{l+1} < p_l$. This means that always some samples migrate from the poor prognosis group to the good one and viceversa. The constrain $p_{l+1} < p_l$ excludes all the genes that leave unchanged the composition of the groups. Here we consider a weaker constraint $p_{l+1} \leq p_l$ instead of the original one in order to include in the signature those genes that support the same group separation. The final signature could result longer than that obtained with the original constrain, but we can further prune the set generated set of genes as reported below.

The main steps to find a signature from a candidate gene follows. Let j be one of the seed genes, selected with the previous step, from which to build the signature.

1. find two groups by using the non missing data in the j^{th} gene and set α to the p -value of the log-rank test; let $\mathcal{S} = \{j\}$ be signature.
2. For each gene j' not in \mathcal{S} , cluster in two groups the dataset by using the features $\mathcal{S} \cup \{j'\}$ and evaluate the corresponding p -value
3. set p_{min} as minimum of the p -values in step (2)
4. if $p_{min} > \alpha$ stop the searching and exit; the signature is in \mathcal{S} .
5. consider those genes \mathcal{S}_{tmp} for which the corresponding p -value is p_{min}
6. if the groups obtained by clustering the genes $\mathcal{S} \cup \mathcal{S}_{tmp}$ are consistent and have tolerably balanced dimensions, then set \mathcal{S} to $\mathcal{S} \cup \mathcal{S}_{tmp}$, α to p_{min} and continue from step (2); otherwise set p_{min} as minimum of the p -values in step (2) except those corresponding to the genes in \mathcal{S}_{tmp} and continue from step (4)

The classification method adopted for finding the clusters is the Partitioning Around Medoids (PAM) from Kaufman and Rousseeuw [12] which falls in the class of K -medians. Other approaches more specific of biological data are also possible [5].

This method can handle missing data and is generally robust to outlying data, although some problem can be meet when a value is so far away from the bulk to be identified itself as a cluster.

Statistical analysis of the signature and pruning. For each gene of a signature, we measure its importance with respect to the other genes and we test if the gene is differentially expressed; when this last event occurs we can

provide also information on how weak or strong is the differential regulation. This information is particularly useful for further biological analysis to be performed *in vivo*.

Let \mathcal{S} be a signature whose elements are the genes g_1, g_2, \dots, g_L , where L is the length of the signature; let $p(\mathcal{S})$ be the p-value provided by the log-rank test for the signature \mathcal{S} and $p(\mathcal{S}^{(l)})$ be the p-value of the signature $\mathcal{S}^{(l)}$ obtained from \mathcal{S} leaving out the l^{th} gene, $l = 1, 2, \dots, L$. The measure of importance of the l^{th} gene with respect to the signature is

$$i(g_l | \mathcal{S}) = \text{Log} \frac{p(\mathcal{S}^{(l)})}{p(\mathcal{S})}, \quad l = 1, 2, \dots, L.$$

The ratio $p(\mathcal{S}^{(l)})/p(\mathcal{S})$ measures the separation between the good and poor prognosis groups with when g_l is left out from the signature \mathcal{S} with respect to the separation measured from the complete signature. As smaller is the separation between the groups as higher is the intensity of the index. It can happen that $p(\mathcal{S}^{(l)}) < p(\mathcal{S})$. This is the case when the omission of the gene improves the p-value.

Once that a given signature has been generated, in order to effectively use it in prognostic case, one should give the indication of the expected behavior of a given gene, *i.e.* it should be differentially regulated in the various prognostic groups. Therefore, in addition to the above indicator, we further apply two statistical controls to the signature in order to check if its members, the genes, are differentially regulated. The first is the test about the differential expression of the gene with respect to good and poor prognosis groups. This test if performed through the standard permutation test methodology [21] concerning the Welch's test statistic which is equivalent to the t -test when the variances of the two population are unequal. The rejection of equal level of gene-expression hypothesis means that the gene can be found differentially expressed in subsequent empirical experiment. The second statistical control we propose is closed to the first but provide a different information. Two bootstrap confidence intervals at 95% level are computed for the expressions of the gene in the two groups. We calculate these only when the previous test rejects the hypothesis of equal means. The confidence intervals can be overlapping or not. When they overlaps we conclude that the expression-gene difference is weak, otherwise if the intervals have no intersection we consider the expression-gene as strong. Each signature can be pruned with some rule concerning the gene-importance ans/or by taking into account the results of the equal means hypothesis testing. Our proposed strategy is to prune the signatures by using the results of the test in view of a later use of the genes for prognosis evaluation.

Gene ranking and integrated signature. An overall signature can be extracted from $\cup_k \mathcal{S}_k$ where \mathcal{S}_k , $k = 1, 2, \dots, K$, is a signature each of one stemmed from a seed gene. The selection of the genes can be done with respect to some indicators. The first one is n_l that counts how many signature \mathcal{S}_k contain the gene g_l . A second indicator is the *mean importance* $i(g_l)$ of the gene g_l , defined as

$$i(g_l) = \frac{1}{n_l} \sum_{\{\mathcal{S}: g_l \in \mathcal{S}\}} i(g_l | \mathcal{S}),$$

and its weighted version

$$I(g_l) = \frac{n_l}{K} i(g_l)$$

we call *weighted importance*.

These indicators allow to rank the genes in $\cup_k \mathcal{S}_k$ and select a subset providing a meta-signature. This is an improvement of the greedy procedure of [3] where the sequential nature of the process tends to generate suboptimal solutions due to local minima of the optimality criterion. Furthermore the gene ranking has the advantage to be easily interpreted by life scientists.

As an example, our procedure generates 16 signatures as reported in table 2. Our final signature is obtained by selecting all the genes of table whose indicators n_l , $i(g_l)$ and $I(g_l)$ are above their corresponding averages and whose expression values are significantly different in the diagnostic group according to the permutation test described above.

3 Results

The data consider for the evaluation of the proposed algorithm is the Non Small Lung Cancer Dataset reported in [14]. The study contains the normalized expression values of 158 genes observed in 148 patients treated by patients treated by lobectomy or pneumonectomy. The set of potential prognostic genes were chosen on the basis of a set of microarray experiments reported by the authors and using further biological knowledge about genes associated with poor prognosis in gene with KRAS mutations. The expression of the 158 candidate gene was measured by using with RT-qPCR [14].

In this section first we analyze the signature obtained with the mSD algorithm, then we run our proposed algorithm on the gene expression data.

3.1 Analysis of the Signature of Boutros et al.

The original algorithm mSD found a signature of 6 genes (STX1A, HIF1A, CCT3, HLA-DBP1, MAFK and RNF5) with a p -value of $2.14 \cdot 10^{-8}$. When the same set of genes is processed by PAM, the corresponding separation of the survival curves has $5.5 \cdot 10^{-4}$ as p -value.

The statistics evaluated on this signature are in table 1. The first column contains the percentage of missing data and no critical situation, on the multi-dimensional point of view, has to be highlighted. In the second column there is the measure of the importance of each gene.

When STX1A is left out from the signature, the p -value computed by considering the other 5 genes increases up to 0.2789 providing an high importance of this gene $i(STX1A | \text{Boutros}) = 2.704$. In the case of HIF1a the importance is $i(HIF1a | \text{Boutros}) = -0.956$, being $6.097 \cdot 10^{-5}$ the p -value of the signature when HIF1a is omitted. It seems that the absence of this gene provides a better

Table 1. Summary statistics for the signature of Boutros et al. (G) good prognosis group, (P) poor prognosis group; loCL is the lower bound of the confidence interval at 0.05 level, while upCL is the corresponding upper bound.

	missingPct	import	meanDiff	pV	goodIs	(G)Lev	(G)loCL	(G)upCL	(P)Lev	(P)loCL	(P)upCL
STX1A	0	2.704	-10.763	0	down	-0.989	-1.235	-0.751	0.538	0.293	0.752
HIF1a	4.082	-0.956	-8.993	0	down	-0.85	-1.159	-0.562	0.516	0.31	0.707
CCT3	0	-0.456	-6.384	0	down	-0.304	-0.476	-0.151	0.372	0.176	0.548
MAFK	17.007	3.209	-2.847	0.003	down	-0.278	-0.634	0.085	0.217	-0.006	0.413
RNF5(*)	0	2.455	-0.868	0.183							
HLA_DBP1	0	1.161	5.104	0	up	0.394	0.137	0.636	-0.365	-0.61	-0.1

signature, although in its the sequential building HIF1a acts as gene that decreases the *p*-value of the sequence under construction. The third column of table 1 is the mean difference of the expression level of the gene with respect to the good and poor prognosis groups. The following column is the *p*-value concerning the null hypothesis of equal expression level. Also in this case RNF5 is in evidence given that is the only gene of the signature for which the null hypothesis is not rejected. This result seems to be consistent with the lowest importance of this gene. In the use of this signature for prognosis evaluation RNF5 can be left out given that is undifferentiated with respect to the good and poor groups. The columns 6 and 9 are the mean levels of the expression of the genes in the two groups. STX1A, HIF1A, CCT3 and MAFK appear to be down-regulated in the good prognosis group while HLA-DBP1 is up-regulated in the same group. For each level, and for both groups, the bootstrap confidence limits (columns 7-8 and 10-11 of table 1) are computed.

3.2 Analysis with the Proposed Algorithm

The following computation are applied to 153 genes instead of the full data. 5 genes have more than 75% of missing values so they are not considered. About the 64% of the genes have been recognised as showing bimodality, while about the 17% provide a significative (at level 0.05) separation of the population survival curves. At the end of the first step 16 genes (ACTR3, COL1A2, FADD, GPR56, PHB, SELP, SERPIND1, SNRPF, SPRR1B, SSBP1, STARD10, STX1A, TIP47, ZFP, CACNA1L and CALCA) are considered as seeds.

Table 2 shows all the signatures found, and it represents our main result.

The first column contains the gene seed, while in the second the signature is listed and the order of the genes is that generated by the algorithm. In the third column the *p*-value of the signature is shown after a decimal-log transformation. As we can see there are several signatures with a *p*-value lower than $5.5 \cdot 10^{-4}$ as obtained by the standard mSD algorithm. Some of the genes in each signature are closed in round brackets, this mean that in the pruning step those genes are not differentially expressed with respect to the classification induced by the signature at level 0.05. The *p*-value of the signature after the removing of the genes in brackets is in the last column. In two cases (ACTR3 and ZFP) such a

Table 2. Signatures developed from the 16 seed genes. The genes between brackets are not differentially expressed at level 0.05. The p -value in column 3 is evaluated with respect to all genes; the following p^* -value is computed after the removing from the signature of those genes not differentially expressed.

Seed	Signature	$\log(pV)$	$\log(p^*V)$
ACTR3	ACTR3, STX1A, (STARD10), SSBP1, FADD, DOC_1R, CTLA4, SPRR1B, (RELA), PER1	-12.657	-0.904
COL1A2	COL1A2, SERPIND1	-1.313	-1.313
FADD	FADD, STX1A, STC1, (RNF5), LAMB1, RELA, (ID2), (KIAA1128), (MYC), TRIO, (MYH11), ACTR3, PLGL, RAFTLIN, (EP300), MSN, (CTNNND1)	-12.49	-1.819
GPR56	((GPR56), KRT5, (DDC), ICA1	-2.105	-1.522
PHB	(PHB), STX1A, SPRR1B, ZWINT	-5.996	-3.655
SELP	SELP, ADM, PAFAH1B3, PHB, (IL20RA), G22P1, EIF4A2, AKAP12, (ARCN1), ARHGDI, (COL9A2), CSTB, CTLA4, CTNNND1, (DOC_1R), (FEZ2), ID2, KIAA0101, (LYPLA1), (MAFK), (NAP1L1), (NICE_4), PTK7, (RAF1), SNRPB, SPRR1B, STX1A, FOSL1	-6.997	-5.876
SERPIND1	SERPIND1, CALCA, SNRPB, HIF1a, SPRR1B, ZWINT, CNN3, EIF4EL3, (KIAA0905), RNF5, CIT, (ID2), (IL6ST), PAFAH1B3, (RMB5), STARD10, (XBP1), RET, STX1A, NAP1L1, TRA1	-10.804	-8.198
SNRPF	SNRPF, CALCA, WFDC2, GRB7, CCT3, COL9A2, CPE, (CTLA4), EIF4EL3, (KIAA0767), (LTB4DH), MLPH, MORF, PAFAH1B3, (PLGL), (RAF1), THRAP2, WEE1, RET, IGJ, (FEZ2), (MYLK), NYFyb, (SLC20A1), (KTN1)	-7.732	-5.554
SPRR1B	SPRR1B, STX1A, FADD, RAFTLIN, (COPB)	-8.765	-6.887
SSBP1	(SSBP1), CCR7, G22P1, (IL20RA), KTN1, HIF1a, NAP1L1, TEB4, (COL1A1), (RNF5), DDC, (SLC20A1), (ARCN1), (FOXA1), HLA_DPB1, (ID2), (IRX5), PER1, (ACTG2), ACTR3, CIT, CNN3, CPE, (CSTB), (CTLA4), (IL6ST), (KIAA0101), (KIAA0905), (LAMB1), LYPLA1, MAFK, MYH11, (RELA), (RUNX1), SELL, SPRRIB, TRIO	-8.999	-7.088
STARD10	STARD10, STX1A, CPE, CACNA1L, PAFAH1B3, CCR7, (CSTB), (CTLA4), (HNRNPAB), LAMB1, NAP1L1, (RAF1), ZWINT, (SNRPB), HIF1a, TRIO, (LMNB1), EIF4EL3, MYH11, ITPKB, (RAFTLIN)	-12.026	-6.819
STX1A	STX1A, FADD, STC1, (RNF5), LAMB1, RELA, (ID2), (KIAA1128), (MYC), TRIO, (MYH11), ACTR3, PLGL, RAFTLIN, (EP300), MSN, (CTNNND1)	-12.49	-1.819
TIP47	TIP47, (TFF3), COL9A2, CCR7	-4.368	-2.754
ZFP	ZFP, MAD2L1, (THBD), HLA_DPB1, (RNF5)	-3.713	-0.914
CACNA1L	CACNA1L, XBP1	-2.192	-2.192
CALCA	CALCA, SERPIND1, SNRPB, HIF1a, SPRR1B, ZWINT, CNN3, EIF4EL3, (KIAA0905), RNF5, CIT, (ID2), (IL6ST), PAFAH1B3, (RMB5), STARD10, (XBP1), RET, STX1A, NAP1L1, TRA1	-10.804	-8.198

p -value becomes greater than 0.05 (-1.30 after the log-transformation), while the signature developed from COL1A2 has a p -value very closed to the level. Later the signatures starting from ACTR3 and ZFP are no longer considered.

Analysis of STX1A-signature. As an example now we concentrate on the signature expanded from STAX1A that counts 16 more genes (STX1A, FADD, STC1, RNF5, LAMB1, RELA, ID2, KIAA1128, MYC, TRIO, MYH11, ACTR3, PLGL, RAFTLIN, EP300, MSN and CTNNND1) with a p -value of $3.234 \cdot 10^{-13}$. Figure 11 display the empirical survival curves estimated on the signature.

The statistics evaluated on this signature are in table 3. The amount of missing value for each gene is moderate. The inspection of the importances shows that

Table 3. Summary statistics for the signature developed from STX1A. The genes marked with (*) are not differentially expressed at level 0.05 with respect to the good and poor prognosis groups. (G) good prognosis group, (P) poor prognosis group; loCL is the lower bound of the confidence interval at 0.05 level, while upCL is the corresponding upper bound.

	missPct	import	meanDiff	pV	goodIs	(G)Lev	(G)loCL	(G)upCL	(P)Lev	(P)loCL	(P)upCL
STX1A	0	12.077	-11.255	0	down	-0.6	-0.799	-0.412	1.038	0.751	1.257
FADD	0	10.303	-6.257	0	down	-0.204	-0.368	-0.067	0.477	0.27	0.648
STC1	0.68	10.6	-6.168	0	down	-0.504	-0.783	-0.237	0.612	0.291	0.926
RNF5(*)	0	1.48	-0.942	0.186							
LAMB1	0	2.452	-2.76	0.002	down	-0.119	-0.25	0.068	0.198	0.005	0.399
RELA	2.041	10.885	-2.983	0.003	down	-0.133	-0.274	0.026	0.243	0.033	0.472
ID2(*)	0	2.017	0.487	0.312							
KIAA1128(*)	1.361	0	-1.603	0.071							
MYC(*)	0	11.743	-0.684	0.249							
TRIO	3.401	2.034	-2.802	0.002	down	-0.187	-0.348	-0.005	0.148	-0.067	0.369
MYH11(*)	3.401	3.55	0.981	0.169							
ACTR3	0	1.009	-6.289	0	down	-0.209	-0.356	-0.056	0.488	0.3	0.695
PLGL	7.483	4.645	-3.419	0	down	-0.432	-0.65	-0.089	0.243	-0.064	0.624
RAFTLIN	0	2.907	2.252	0.007	up	0.193	0.014	0.381	-0.151	-0.472	0.136
EP300(*)	0	0	0.69	0.25							
MSN	0	0.431	-2.145	0.011	down	-0.102	-0.232	0.046	0.181	-0.074	0.468
CTNNND1(*)	0.68	0	-1.009	0.167							

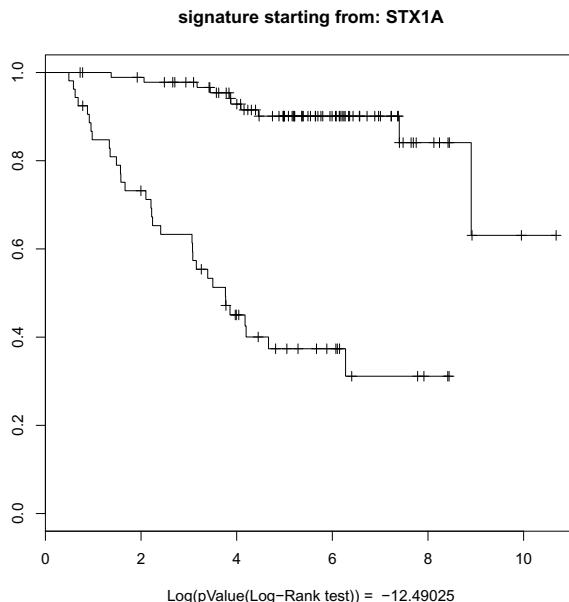


Fig. 1. Display of the empirical survival curves estimated by using the classification provided the signature developed from STX1A

the inclusion of some gene in the sequence leaves the *p*-values unchanged, this is the case of KIAA1128, EP300 and CTNND1. These three genes with lowest importance, together with RNF5, ID2, MYC and MYH11, are not differentially expressed. MYC is worth of some remark. It is the second important gene, after STX1A, but at the same time appear not to be differentially expressed. This seems to suggest that being a strong carrier of information about a good group separation is unconnected with the feature of having different level of expression in the groups. We could hypothesize that some time the bimodality associated to the good and poor prognosis group is a model too simple.

The strong separation of the levels between the groups is for STX1A, FADD, STC1, RELA and ACTR3, while LAMB1, TRIO, PLGL, RAFTLIN and CTNND1 show a weak differential expression. When we restrict our attention to the set of genes having a significative different expression level, the classification provides two groups for which the survival curves have 0.015 as *p*-value.

4 Conclusions and Future Works

We reported a signature learning algorithm from gene expression. The reported results show how our method improves, in terms of the chosen separation measure, the difference between prognostic groups induced by the selected features. We used a real dataset of 158 genes measured in 148 samples measured by RT-qPCR for prognosis of NSCLC.

References

1. Alizadeh, A.A., et al.: Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature* 403(6769), 503–511 (2000)
2. Ambroise, C., McLachlan, G.J.: Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences of the United States of America* 99(10), 6562–6566 (2002)
3. Boutros, P.C., Lau, S.K., Pintilie, M., Liu, N., Sheperd, F.A., Der, D.S., Tao, M., Penn, L.Z., Jurisca, I.: Prognostic gene signatures for non-small-cell lung cancer *Arch. Rat. Mech. Anal.* 78, 315–333 (1982)
4. Cai, Y.D., Huang, T., Feng, K.-Y., Hu, L., Xie, L.: A unified 35-gene signature for both subtype classification and survival prediction in diffuse large B-cell lymphomas. *PloS one* 5(9), e12726 (2010)
5. Ceccarelli, M., Maratea, A.: Improving fuzzy clustering of biological data by metric learning with side information. *International Journal of Approximate Reasoning* 47(1), 45–57 (2008)
6. Chang, H., Nuyten, D., et al.: Robustness, scalability, and integration of a wound-response gene expression signature in predicting breast cancer survival. *PNAS* 102(10), 3738–3743 (2005)
7. Chen, H.-Y., et al.: A Five-Gene Signature and Clinical Outcome in NonSmall-Cell Lung Cancer. *The New England Journal of medicine* 356(1), 11 (2007)
8. Van De Vijver, M.J., et al.: A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medicine* 347(25), 1999–2009 (2002)

9. Golub, T.R., et al.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* 286(5439), 531 (1999)
10. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. *Machine Learning* 46, 389–422 (2002)
11. Jain, A.K., Zongker, D.: Feature Selection: Evaluation, Application, and Small Sample Performance. *IEEE Trans. Pattern Analysis and Machine Intelligence* 19(2), 153–158 (1997)
12. Kaufman, L., Rousseeuw, P.J.: *Finding Groups in Data*. Wiley, Chichester (1990)
13. Lapointe, et al.: Gene expression profiling identifies clinically relevant subtypes of prostate cancer. *Proceedings of the National Academy of Sciences of the United States of America* 101(3), 801 (2004)
14. Lau, S., et al.: Three-gene prognostic classifier for early-stage non-small-cell lung cancer. *Journal of Clinical Oncology* 25(25), 5562–5566 (2007)
15. Lisboa, P., Velido, A., Tagliaferri, R., Ceccarelli, M., Martin-Guerrero, J., Biganzoli, E.: Data Mining in Cancer Research. *IEEE Computational Intelligence Magazine* 5(1), 14–18 (2010)
16. Mantel, N.: Evaluation of survival data and two new rank order statistics arising in its consideration. *Cancer Chemother. Rep.* 50(3), 163–170 (1966)
17. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern. Anal. Mach. Intell.* 27, 1226–1238 (2005)
18. Sørlie, T., et al.: Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences of the United States of America* 98(19), 10869 (2001)
19. Rousseeuw, P.J., van Driessen, K.: A Fast Algorithm for the Minimum Covariance Determinant Estimator. *Technometrics* 41, 212–223 (1999)
20. Schwarz, G.: Estimating the dimension of a model. *Ann. Stat.* 6, 461–464 (1978)
21. Welch, W.J.: Construction of Permutation Tests. *Journal of the American Statistical Association* 85(411), 693–698 (1990)
22. Zhang, X., Qian, X.L., Xu, X.-Q., Leung, H.-C., Harris, L., Iglehart, J., Miron, A., Liu, J., Wong, W.: Recursive SVM feature selection and sample classification for mass-spectrometry and microarray data. *BMC Bioinformatics* 7, 197 (2006)

An Empirical Comparison of Flat and Hierarchical Performance Measures for Multi-Label Classification with Hierarchy Extraction

Florian Brucker, Fernando Benites, and Elena Sapozhnikova

Department of Computer and Information Science, University of Konstanz
FirstName.SecondName@uni-konstanz.de

Abstract. Multi-label Classification (MC) often deals with hierarchically organized class taxonomies. In contrast to Hierarchical Multi-label Classification (HMC), where the class hierarchy is assumed to be known *a priori*, we are interested in the opposite case where it is unknown and should be extracted from multi-label data automatically. In this case the predictive performance of a classifier can be assessed by well-known Performance Measures (PMs) used in flat MC such as precision and recall. The fact that these PMs treat all class labels as independent labels, in contrast to hierarchically structured taxonomies, is a problem. As an alternative, special hierarchical PMs can be used that utilize hierarchy knowledge and apply this knowledge to the extracted hierarchy. This type of hierarchical PM has only recently been mentioned in literature. The aim of this study is first to verify whether HMC measures do significantly improve quality assessment in this setting. In addition, we seek to find a proper measure that reflects the potential quality of extracted hierarchies in the best possible way. We empirically compare ten hierarchical and four traditional flat PMs in order to investigate relations between them. The performance measurements obtained for predictions of four multi-label classifiers ML-ARAM, ML- k NN, BoosTexter and SVM on four datasets from the text mining domain are analyzed by means of hierarchical clustering and by calculating pairwise statistical consistency and discriminancy.

Keywords: Multi-label Classification, Text Classification, Performance Measures, Hierarchical Classification.

1 Introduction

A common way to structure large amounts of information is the use of a hierarchical taxonomy, which describes relations between classes or concepts by means of the notion “is-a”. With the ever-increasing amount of data, automatic classification of items into such taxonomies is often beneficial or even inevitable. This classification becomes multi-label because each object belongs not only to a single class but also to all its ancestors (ancestor inclusion property). One of

the important research fields in Multi-label Classification (MC) is hierarchical multi-label classification (HMC) where the hierarchical taxonomy is assumed to be known a priori and is available for a classifier together with labeled data. Although a lot of information can be assumed to be hierarchically structured, sometimes the hierarchical taxonomy is not explicitly given but can be derived from data by analyzing the multi-labels. In this much less investigated setting, the classifier obtains multi-label data without information on class hierarchy and performs multi-label classification. Additionally a Hierarchy Extraction (HE) algorithm can extract a class hierarchy from the training multi-labels. In this setting, the predictive performance of a multi-label classifier can be assessed either by well-known Performance Measures (PMs) used in flat multi-label classification such as precision and recall or by special hierarchy measures that utilize hierarchy knowledge, which have only recently been mentioned in literature. In the latter case the extracted hierarchy can be used, however its quality depends on the choice of HE algorithm that is used.

Performance evaluation by flat measures in the hierarchical taxonomy case has been shown to be disadvantageous [1]: In traditional MC, there is no direct relation between individual labels, and therefore the corresponding PMs treat them equally. With a hierarchical taxonomy, the relations between labels are given by the hierarchy in the form of ancestor inclusion and should be taken into account when evaluating the quality of classification results. If a certain algorithm classifies an article about the soccer world championship into the category *sports* instead of *soccer*, for example, then this should be considered a smaller error in comparison with classifying the article under *economy*, as the true label, *soccer* is a descendant of *sports*. Thus, the distance between predicted and true labels in the hierarchy should be measured in addition to simply counting misclassified labels. On the other hand, correct predictions of more general categories corresponding to higher levels in the hierarchy and being made instead of specific ones are usually less interesting because the deeper a label in the hierarchy the more difficult is to predict it due to the lack of training examples and the more valuable its prediction.

The fact that both issues can be addressed in a variety of ways has resulted in the development of a large number of special hierarchical measures. In the past ten years (1999-2009), at least ten PMs for HMC have been introduced in the literature by various authors, but none of them has been established yet as a standard one. Due to the large number of existing measures and the lack of comparative studies in the field it is difficult to choose a proper measure or a set of measures for performance comparisons. Using several measures generally provides multiple views on the results, therefore giving the user more information to make decisions, however the selection of PMs must be made very carefully. Many measures may produce similar results even though their approaches initially appear to be different. It is therefore important to know how the individual measures are related to each other to ensure that the chosen set of measures is, in a certain sense, independent.

The aim of this paper is a comparative analysis of the existing PMs in order to facilitate the search for the proper combination of PMs. To investigate relations between measures we use predictions of four multi-label classifiers ML-ARAM, ML- k NN, BoosTexter and SVM on four datasets from the text mining domain. Since we are interested in Hierarchy Extraction (HE), we exploit non-hierarchical classifiers. To compare the measures, we apply cluster analysis and the approach proposed in [9] for comparison of two PMs on the basis of statistical consistency and discriminancy. In this context three issues will be addressed. The first issue concerns the fact that flat measures have been studied extensively and their behavior is well understood. This is not the case for most of the hierarchical PMs, and often their rather complex definitions do not allow for an intuitive understanding of their behavior. To solve this problem, we will empirically compare hierarchical and flat measures in order to associate the hierarchical measures with the flat measures that tend to give similar results. The second issue involves verifying that the hierarchical measures do indeed judge the severity of a misclassification, providing more details than the flat ones. The third issue concerns HE. It is often difficult to determine whether the predicted multi-labels can produce a good hierarchy or not. A quantitative comparison of extracted and original hierarchies by means of hierarchy proximity measures can be used as a quality measure for predictions once the original hierarchy is known. In the opposite case, the PM to be used should correlate sufficiently with the quality of the extracted hierarchies. Unfortunately, it has been found [4] that the flat measures and one hierarchical measure do not possess this desirable property. A more comprehensive analysis will be provided: HMC PMs will be examined with respect to HE in order to find the best indicator of HE quality among them.

To the best of the authors' knowledge, only little related work has been done to date. In [7] the authors provide theoretical arguments for the choice of PMs in flat MC. In [11] Kiritchenko defines the three theoretical properties a good HMC PM should possess. The work in [6] contains an overview of PMs for single-label hierarchical classification. Finally, in [13] the authors present an overview of hierarchical classification methods and provide a short overview of a number of HMC PMs. We point out that there are apparently no studies as yet that provide an empirical comparison of hierarchical measures. Furthermore there is no consecutive study comparing hierarchical and flat PMs. Our work aims at filling this deficiency with experimental results from real-world data.

The paper is organized as follows. In Sec. 2 we introduce the necessary concepts and notation. The comparisons between the hierarchical and flat measures as well as the analysis of relationships between HMC PMs and HE are given in Sec. 3. Finally, Sec. 4 concludes the paper.

2 Concepts and Notation

In traditional MC, the task is to learn a mapping $f : X \rightarrow \mathcal{P}(L)$ from the set of instances X to the set of multi-labels $\mathcal{P}(L)$. Here, a *multi-label* y is any subset of the *label set* $L = \{1, \dots, q\}$ and \hat{y} is the classifier prediction. MC PMs are based

Table 1. Overview of the applied HMC PMs with respective citations, ranges, and groups. “...” means that the measure has an unrestricted range

Measure	F_K	F_V	F_I	F_S	D_C	D_{CB}	D_S	D_{W1}	S_N	D_{W2}
Reference	[11]	[17]	[10]	[15]	[3]	[4]	[14]	[20]	[12]	[18]
Range	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0, ...]	[0, ...]	[0,1]	[0,1]	[0,1]
Group	F-measures		Hamming Distance			label-to-label				

on the calculation of binary classification counters: the numbers of true positives, false positives, true negatives and false negatives. The most popular measures include *accuracy*, *precision*, *recall*, and the *F-measure* [16]. There are also two possible methods to average the measures over a dataset: macro-averaging assigns equal weight to each label, while micro-averaging assigns equal weight to each multi-label [21]. Here, we use only the micro-averaged versions.

The *accuracy* A of the predictions refers to the fraction of correctly predicted labels. The *precision* P measures how many of the predicted labels are actually present, the *recall* R measures how many of the present labels have been predicted, and the *F-measure* is the harmonic mean of precision and recall: $F := \frac{2PR}{P+R}$.

All of these measures take values in $[0, 1]$, with higher values indicating better predictions. In addition to these label-based measures, instance-based ones can be used, when one is interested in an instance-wise performance measurement: $P_{in} := \frac{|y_s \cap \hat{y}_s|}{|\hat{y}_s|}$, $R_{in} := \frac{|y_s \cap \hat{y}_s|}{|y_s|}$. We use them later for pairwise comparisons of HMC performance measures by means of the degree of consistency and the degree of discriminancy.

HMC extends MC by imposing a hierarchical structure on the label set L : A *hierarchy* H on L is a set of pairs (p, c) where $p, c \in L$ and p is considered a parent of c . Here we suppose a hierarchy to be a tree. As we exploit global non-hierarchical classifiers, the information on the hierarchy is available to the classifier in an indirect way, i.e. as provided by multi-labels. Note that while it is assumed that the true multi-labels are consistent with the hierarchy, this is usually not true for the predicted multi-labels since they are affected by classifier performance.

An *HMC PM* is a proximity measure between sequences of multi-labels that takes into account a hierarchy H . The goal is to quantify how close the predictions are to the true multi-labels. The HMC PMs suggested so far in the literature can be roughly divided into three groups: The first group consists of hierarchical versions of the popular F-measure (denoted by a capital F), the second one contains hierarchical versions of the Hamming distance, and measures in the third group are based on label-to-label distances. Another categorization divides HMC PMs into *distances*, denoted by a capital D , where lower values indicate better results, and *similarities*, denoted by a capital S , where the opposite relation is the case. Table II shows the list of HMC measures used for comparison. It should be noted that the range of some measures is not be generally restricted, however in specific cases the range is bound applied hierarchy and the dataset.

Table 2. Datasets overview

Name	depth	cardinality	labels	# feat.	paths	instances	train	test
RCV1-v2	4	3.18	103	945	1.44	23,149	15,000	8,149
WIPO-alpha	4	4.32	131	924	1.20	12,052	4,688	7,364
ASRS	2	5.01	67	462	3.08	15,000	10,000	5,000
OHSUMED	4	1.92	101	592	4.32	16,086	12,428	3,658

Most of the measures are based on the calculation of a multi-label to multi-label difference from which the overall performance measure for a whole dataset is derived as the average difference per multi-label. For F_S we set the parameter $Dis_\theta := 1$.

3 Comparison of PMs

To compare the PMs, each of them was applied to the classification results taken from four multi-label classifiers obtained from four datasets. The classifiers were the neural-network-based ML-ARAM, the nearest-neighbor based ML- k NN, the rule-based BoosTexter, all three used as in [2], and the Support Vector Machine (SVM) implementation LIBSVM [5]. For SVM, label combination and c-svc were used.

The characteristics of the used datasets are summarized in Table 2. Depth corresponds to the maximal number of levels in the class hierarchy. Cardinality is the average number of labels per instances. The paths column means how many paths a multi-label in the dataset has on average. The data were randomly divided into training and test set, if not already divided in the original work.

The RCV1-v2 dataset was used as in [2] and the WIPO-alpha dataset as in [1]. The Aviation Safety Reporting System (ASRS) dataset consists of anonymous reports of flight incidents [19]. During preprocessing, stop-words were removed and remaining words were stemmed using the Snowball stemmer. All but the 2%-most frequent stems were removed. Conversion to TF-IDF¹ weights and normalization was done independently for the training and the test set. The OHSUMED dataset was used here as in [8], except that the root node of the hierarchy, with the maximal depth of four, and the documents assigned to it were not used. The preprocessing was the same as for the ASRS dataset.

3.1 Cluster Analysis

The first approach we used to compare of PMs was hierarchical clustering. It groups similar objects together by building a hierarchy of clusters which become more and more coarse from the bottom levels to top levels. The idea was to visualize the relations between individual HMC PMs as well as between them and the flat PMs. In this experiment the original hierarchy was used to calculate

¹ Term Frequency Inverse Document Frequency

hierarchical measures. Using predictions of four classifiers on four datasets yields 16 classifier/dataset combinations, which are considered to be a vector $v \in \mathbb{R}^{16}$, for each measure. The similarity between two PMs D_1 and D_2 can subsequently be defined as

$$s(D_1, D_2) := \frac{1 + \rho(v_{D_1}, v_{D_2})\sigma(D_1)\sigma(D_2)}{2}. \quad (1)$$

Here, $\rho(X, Y)$ is Spearman's rank correlation coefficient, which is close to 1 (-1) if the relationship between X and Y can be described well using a monotonically increasing (decreasing) function, not necessarily linear. If $\rho(X, Y)$ is close to 0, then no monotone relation between X and Y exists. $\sigma(D) := 1$ for similarities and $\sigma(D) := -1$ for distances. Thus, $s(D_1, D_2)$ is close to 1 if D_1 and D_2 agree on which of two given classification results is better, and close to 0 if they disagree. A value of $s(D_1, D_2)$ close to 0.5 indicates that there is no clear relationship between both measures.

Applying hierarchical clustering with average linkage to the similarity values calculated according to (1) results in the dendrogram shown in Fig. 1. For a medium threshold (0.95), the following five clusters are obtained, with some of them being close to the flat measures: D_C , D_{W2} , and D_S are close to the accuracy; D_{CB} ; F_K , S_N , F_V , and F_S are close to the recall and the F-measure; D_{W1} ; F_I .

It is important to note that these clusters do not correspond to the intuitive categorization of the HMC measures into three groups presented above. For example, F_I is not in the same cluster as all the other hierarchical F-measures. This is an interesting point because it confirms the hypothesis that the measure selection based exclusively on intuition may lead to biased interpretation of classification results. Since measures in the same cluster behave similarly, it is advisable to use only one measure from each cluster.

3.2 Relation to Hierarchy Extraction (HE)

In [2] the task of HE is defined as constructing an appropriate hierarchy H based on a given a set of predicted multi-labels. The authors propose to measure the

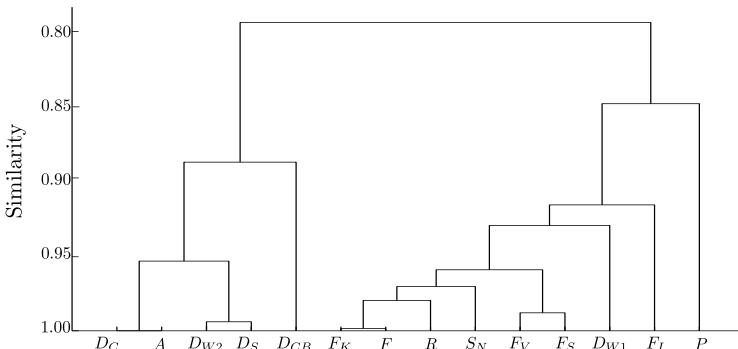


Fig. 1. Dendrogram for average linkage hierarchical clustering of PMs using similarities

Table 3. Similarities between HE metrics and other HMC metrics as measured by [10]

	F_K	F_V	F_I	F_S	D_C	D_{CB}	D_{W1}	S_N	D_{W2}	D_S
D_{CTED}	0.86	0.86	0.87	0.85	0.62	0.54	0.93	0.91	0.77	0.74
D_{TO^*}	0.85	0.88	0.91	0.86	0.59	0.49	0.90	0.91	0.74	0.70
D_{LCAPD}	0.88	0.88	0.90	0.88	0.62	0.51	0.93	0.91	0.77	0.74
Avg. Similarity	0.86	0.87	0.90	0.86	0.61	0.52	0.92	0.91	0.76	0.73

Table 4. Average relative difference (in percent) between using original hierarchies and extracted hierarchies when calculating the HMC PMs

	F_K	F_V	F_I	F_S	D_C	D_{CB}	D_{W1}	S_N	D_{W2}	D_S
RCV1-v2	0.0	0.0	0.0	0.0	0.0	10.1	0.0	0.0	0.0	0.2
WIPO-alpha	0.0	0.0	0.0	0.0	0.0	1.1	3.3	0.3	0.1	0.0
ASRS	2.0	4.2	5.9	0.0	0.2	14.7	10.6	5.8	0.2	2.0
OHSUMED	0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0

quality of the extracted hierarchies by comparing them to the real hierarchies using hierarchy proximity measures. Specifically, three distance metrics for trees were used: D_{CTED} based on *Constrained Tree Edit Distance* (CTED), D_{TO^*} based on *Taxonomic Overlap* (TO*), and D_{LCAPD} based on *Lowest Common Ancestor Path tree Distance* (LCAPD) (see [2] for details). Further, in [11] it was observed that neither the flat PMs nor D_{CB} are consistent with the results of hierarchy proximity measures. Nevertheless a PM that is consistent with HE results would be advantageous because such a measure would allow a conclusion to be drawn about the potential quality of HE on the basis of classification results. To examine how the studied HMC PMs are related to the hierarchy proximity measures, the similarity values between them were calculated by [10] (Table 3). D_{CB} yields the lowest average similarity that confirms the result of [11]. In contrast, D_{W1} , S_N , and F_I exhibit the highest similarity to the hierarchy proximity measures. Thus one can successfully use them as indicators for the potential quality of HE.

Another important question concerns the application of HMC PMs to multi-label data for which no hierarchy is given. If one wants to use HMC PMs to evaluate classification experiments on such datasets, a hierarchy extracted by HE techniques from the training multi-labels should be utilized. To test whether this approach yields useful results we extracted hierarchies from the training multi-labels of the four datasets and used the extracted hierarchies to calculate the HMC PMs. Table 4 shows the relative differences between using the original and extracted hierarchies. The quality of the results with the use of extracted hierarchies obviously depends on the quality of the extracted hierarchies. This can be observed by comparing the results for the different datasets: Those for RCV1-v2, WIPO-alpha and OHSUMED are, in general, much better than those for ASRS. Indeed, the tree distance metrics used by [11] show that the original hierarchies could be recovered almost completely for RCV1-v2, WIPO-alpha and OHSUMED, while the extracted hierarchy for ASRS was of lower quality.

Table 5. Average over the four datasets of approximate Degree of Consistency DoC . Note that $DoC(D_1, D_2) = DoC(D_2, D_1)$. The unshaded cells have a standard deviation of less than 0.1, around 0.1 for the light gray and 0.2 for dark gray cells.

D_1/D_2	A	R	P	F
F_I	0.89	0.91	0.90	0.92
D_{W1}	0.83	0.72	0.88	0.87
S_N	0.89	0.84	0.93	0.94

Table 6. Approximate degree of discriminancy $DoD(D_1, D_2)$ between HMC PMs D_1 and flat PMs D_2 . Note that $DoD(D_1, D_2) = 1/DoD(D_2, D_1)$.

D_1/D_2	A	R	P	F	A	R	P	F	
RCV1-v2					WIPO-alpha				
F_I	0.07	0.66	0.13	0.03	0.04	0.14	0.04	0.02	
D_{W1}	11.49	67.28	20.97	4.5e+05	27.44	174.54	49.63	∞	
S_N	10.36	61.43	17.47	5.26	0.85	5.86	1.04	0.35	
ASRS					OHSUMED				
F_I	1.07	1.65	0.82	0.50	0.39	2.44	0.43	0.20	
D_{W1}	34.34	50.44	29.94	∞	7.43	26.21	10.57	∞	
S_N	45.57	38.95	22.32	32.94	43.17	122.27	51.69	296.39	

Another important result is that some HMC PMs are affected more than others when extracted hierarchies are used. The value of F_S , for example, is not affected at all, whereas the one of D_{CB} is changed for ASRS.

3.3 Degree of Consistency and Degree of Discriminancy

In [9], Huang and Ling suggest formal criteria for comparing two PMs and finding out which of them is better. They introduce the *Degree of Consistency* DoC , which shows how strongly two measures tend to agree and the *Degree of Discriminancy* DoD , which shows which of both measures is more discriminative. For two measures D_1 and D_2 , D_1 is said to be *statistically consistent with and more discriminating than* D_2 if $DoC(D_1, D_2) > 0.5$ and $DoD(D_1, D_2) > 1$. Huang and Ling argue that, in this case, D_1 is a better measure than D_2 , since D_1 and D_2 agree most of the time but D_1 has more discriminating power than D_2 .

We applied this approach to find out whether hierarchical PMs do judge the severity of a misclassification with more nuances than the flat ones. To simplify presentation, we focus on three PMs selected above as good indicators of HE quality D_{W1} , S_N , and F_I . The comparison of their DoC values shows a high agreement with the flat measures (Table 5). By examining their degrees of discriminancy in Table 6 one can see that two of them, namely D_{W1} , and S_N indeed possess more discriminating power than any of the given flat measures. In contrast, F_I does not achieve higher discriminancy and therefore we recommend to use D_{W1} and S_N for performance comparisons.

Comparing the classifiers by means of the selected PMs allows one to draw justified conclusions about their predictive power. The values of D_{W1} and S_N for

Table 7. Classification results as evaluated by the optimal combination of HMC PMs with extracted hierarchy. The best result with respect to each measure/dataset/classifier combination is marked in bold.

	<i>ARAM kNN Boost. SVM</i>				<i>ARAM kNN Boost. SVM</i>			
	RCV1-v2				WIPO-alpha			
D_{W1}	0.189	0.280	0.224	0.130	0.264	0.423	0.305	0.093
S_N	0.932	0.893	0.917	0.951	0.909	0.885	0.912	0.931
	ASRS				OHSUMED			
D_{W1}	0.404	0.464	0.393	0.353	0.281	0.444	0.344	0.226
S_N	0.829	0.800	0.816	0.819	0.887	0.832	0.871	0.908

each classifier and dataset are given in Table 7. It is obvious that SVM performs best, receiving the highest scores in all but one of eight comparisons. ML-ARAM slightly outperforms BoosTexter, winning six of the eight direct comparisons between both classifiers. Finally, ML-*k*NN shows the worst performance.

4 Conclusion

In this work, ten hierarchical PMs recently proposed by various authors were empirically compared with four flat ones by applying them to MC with HE. In our setting no original hierarchy is available to a classifier, but can be derived from multi-labels. Based on the predictions of four multi-label classifiers ML-ARAM, ML-*k*NN, BoosTexter and SVM obtained from four datasets from the text mining domain, the performance measures were empirically tested for similar behavior which can lead to biased results. Additionally, hierarchical PMs were to discern their relation to the results obtained from HE. Some of them showed a high correlation with the quality of extracted hierarchies as measured by hierarchy proximity measures. They can therefore serve not only as classification performance measures but also as indicators of the HE performance. The values of the PMs were not affected by using extracted hierarchies instead of the original hierarchy. From the experiments we conducted, we can recommend two measures D_{W1} and S_N . Nevertheless, it is important to point out that the results of this work are not applicable as a stand-alone argument for selecting a suitable set of HMC PMs for a given dataset. Other important aspects regarding the choice of PMs are, for example, their computational complexity or the difficulty of interpreting their results. In addition, some measures may be better suited for certain datasets than others. It is also advisable to use measures that have been used in similar experiments before, in order to simplify the comparison.

When comparing hierarchical PMs with traditional flat ones, the former PMs were shown to improve the quality assessment for MC results obtained on the datasets with hierarchical class taxonomies. At the same time, HMC and MC PMs do in general agree on whether a classification result is good or not. This implies that the HMC measures achieve the intended improvements not only in theory, but also in practice.

References

1. Benites, F., Brucker, F., Sapozhnikova, E.: Multi-Label Classification by ART-based Neural Networks and Hierarchy Extraction. In: Proc. of the IEEE IJCNN 2010, pp. 2788–2796. IEEE Computer Society, Barcelona (2010)
2. Brucker, F., Benites, F., Sapozhnikova, E.: Multi-label classification and extracting predicted class hierarchies. *Pattern Recognition* 44(3), 724–738 (2011)
3. Cai, L., Hofmann, T.: Exploiting known taxonomies in learning overlapping concepts. In: Proc. of Int. Joint Conf. on Artificial Intelligence (2007)
4. Cesa-Bianchi, N., Gentile, C., Zaniboni, L.: Hierarchical classification: combining Bayes with SVM. In: Proc. of the 23rd Int. Conf. on Machine learning (2006)
5. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), <http://www.csie.ntu.edu.tw/~cjlin/libsvm> (acc. 03.2010)
6. Costa, E., Lorena, A., Carvalho, A., Freitas, A.: A review of performance evaluation measures for hierarchical classifiers. In: Proc. of the AAAI 2007 Workshop: Evaluation Methods for Machine Learning II, pp. 1–6 (2007)
7. Davis, J., Goadrich, M.: The relationship between precision-recall and ROC curves. In: Proc. of the 23rd Int. Conf. on Machine Learning, p. 240. ACM, New York (2006)
8. Granitzer, M.: Hierarchical text classification using methods from machine learning. Master's thesis, Graz University of Technology (2003)
9. Huang, J., Ling, C.: Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 299–310 (2005)
10. Ipeirotis, P., Gravano, L., Sahami, M.: Probe, Count, and Classify: Categorizing Hidden-Web Databases. In: Proc. of the 2001 ACM SIGMOD Int. Conf. on Management of Data, pp. 67–78 (2001)
11. Kiritchenko, S.: Hierarchical text categorization and its application to bioinformatics. Ph.D. thesis, University of Ottawa Ottawa, Ont., Canada (2006)
12. Nowak, S., Lukashevich, H.: Multilabel classification evaluation using ontology information. In: Proc. of the First ESWC Workshop on Inductive Reasoning and Machine Learning on the Semantic Web, Heraklion, Greece (2009)
13. Silla, C., Freitas, A.: A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 1–42 (2010)
14. Struyf, J., Dzeroski, S., Blockeel, H., Clare, A.: Hierarchical multi-classification with predictive clustering trees in functional genomics. In: Bento, C., Cardoso, A., Dias, G. (eds.) EPIA 2005. LNCS (LNAI), vol. 3808, pp. 272–283. Springer, Heidelberg (2005)
15. Sun, A., Lim, E.: Hierarchical text classification and evaluation. In: Proc. of the 2001 IEEE Int. Conf. on Data Mining, California, USA, vol. 528 (2001)
16. Tan, P., Steinbach, M., Kumar, V.: Introduction to data mining. Pearson Addison Wesley, Boston (2006)
17. Verspoor, K., Cohn, J., Mniszewski, S., Joslyn, C.: A categorization approach to automated ontological function annotation. *Protein Science* 15(6), 1544–1549 (2006)
18. Wang, K., Zhou, S., He, Y.: Hierarchical classification of real life documents. In: Proc. of the 1st (SIAM) Int. Conf. on Data Mining, pp. 1–16 (2001)
19. Woolam, C., Khan, L.: Multi-concept document classification using a perceptron-like algorithm. In: WI-IAT 2008: Proc. of the 2008 IEEE/WIC/ACM Int. Conf. on Web Intelligence and Intelligent Agent Technology, pp. 570–574. IEEE Computer Society, Washington, DC, USA (2008)

20. Wu, F., Zhang, J., Honavar, V.: Learning Classifiers Using Hierarchically Structured Class Taxonomies. In: Zucker, J.-D., Saitta, L. (eds.) SARA 2005. LNCS (LNAI), vol. 3607, pp. 313–320. Springer, Heidelberg (2005)
21. Yang, Y.: An evaluation of statistical approaches to text categorization. Information Retrieval 1(1), 69–90 (1999)

5 Remarks to the Reviewers Comments

5.1 Reviewer One

Keywords are provided now. The critic point about the weak scientific contribution can not be directly addressed, since it is a vague critic, our subject was already mentioned as important and more investigations have been also requested in literature, i.e. [13]. The main objective of this paper is to provide a base method to choose hierarchical performance measures with the aim to help HMC researchers use only a few measures to evaluate their results and create a base for discussions. At the moment HMC researchers are using very different methods and therefore the results are only of limited comparability.

5.2 Reviewer Two

The language was revised by a British native speaker and her proposals were included. Some examples of changes follows:

- “each of them was applied to classification results of four multi-label classifiers obtained on four datasets” → “each of them was applied to *the* classification results *taken from* four multi-label classifiers obtained *from* four datasets”
- “the range of some measures is not be generally bounded, but specifically the used hierarchy and dataset bound it” → “the range of some measures is not be generally *restricted*, however *in specific cases the range is bound applied hierarchy and the dataset.*”
- “macro-averaging gives equal weight to each label, while micro-averaging gives equal weight to each multi-label [21]. We use only the micro-averaged versions here.” → “macro-averaging *assigns* equal weight to each label, while micro-averaging *assigns* equal weight to each multi-label [21]. *Here*, we use only the micro-averaged versions.”
- “The work [6] contains” → “The work *in* [6] contains”
- “a quality measure for the predictions when the original hierarchy is known.” → “a quality measure for the predictions *once* the original hierarchy is known.”

Towards a Parallel Approach for Incremental Mining of Functional Dependencies on Multi-core Systems

Ghada Gasmi, Yahya Slimani, and Lotfi Lakhal

Department, Faculty of Sciences of Tunis, University of Tunis El Manar, Tunisia
Laboratoire d’Informatique Fondamentale de Marseille (LIF), Aix-Marseille,
Université IUT d’Aix en Provence, France
`{ghada.gasmi,Amira.Beji}@gmail.com, yahya.slimani@fst.rnu.tn`

Abstract. A general assumption in all existing algorithms for mining functional dependencies is that the database is static. However, real life databases are frequently updated. To the best of our knowledge, the discovery of functional dependencies in dynamic databases has never been studied. A naïve solution consists in re-applying one of the existing algorithms to discover functional dependencies holding on the updated database. Nevertheless, in many domains, where response time is crucial, re-executing algorithms from the scratch would be unacceptable. To address this problem, we propose to harness the multi-core systems for an incremental technique for discovering the new set of functional dependencies satisfied by the updated database. Through a detailed experimental study, we show that our parallel algorithm scales very well with the number of cores available.

Keywords: Multi-core systems, data mining, emergent architectures, parallel programming, incremental mining of functional dependencies.

1 Introduction

Originally, the study of functional dependencies (FDs) has been motivated by the fact that they could express constraints holding on a relation independently of a particular instance [1]. Later, Mannila and Rih studied FDs with a data mining point of view. Indeed, the idea was introduced in [9] as the inference of functional dependencies problem. Its principle consists in determining a cover of all functional dependencies holding on a given relation r . Motivations for addressing functional dependencies inference arise in several areas [10,3,5,6,14,12,15,7,2]. Indeed, FDs were applied in database management [4,8], data reverse engineering [13], query optimization and data mining [5]. A crucial study of the dedicated literature allows one to note that a general assumption in all existing algorithms is that the database is static. However, real life databases are dynamic where they are constantly updated. Hence, a possible solution consists in re-applying one of the existing algorithms on the updated database. This solution though simple, has disadvantages. All previous computation done to discover FDs is wasted and the process of FDs

discovery must restart from the scratch. Indeed, the more the size of the database and the frequency of its update increase, the more this solution becomes time consuming and unacceptable in many applications.

To address this problem, we harness multi-core systems to propose an incremental algorithm, called MT-INCFDs, which reuses previously mined FDs and combine them with the new tuples to insert into r in order to efficiently compute the new set FDs. Indeed, we have been inspired by the following two advances. The first innovation is to place multiple cores on a single chip, called Chip Multiprocessing (CMP). Each core is an independent computational unit, allowing multiple processes to execute concurrently. The second advancement is to allow multiple processes to compete for resources simultaneously on a single core, called simultaneous multithreading (SMT). By executing two threads concurrently, an application can gain significant improvements in effective instructions per cycle (IPC). These advances arise in part because of the inherent challenges with increasing clock frequencies. The increase in processor frequencies over the past several years has required a significant increase in voltage, which has increased power consumption and heat dissipation of the central processing unit. In addition, increased frequencies require considerable extensions to instruction pipeline depths. Finally, since memory latency has not decreased proportionally to the increase in clock frequency, higher frequencies are often not beneficial due to poor memory performance. Hence, by combining CMP and SMT technology, chip vendors can continue to improve IPC without raising frequencies. In order to highlight the benefit of using multi-core systems, we conducted a set of experiments on synthetic databases.

The remainder of the paper is organized as follows. Section 2 presents some key notions used throughout the paper. In section 3, we introduce the MT-INCFDs algorithm for incremental mining of FDs on CMP systems. Experimental evaluations are given in Section 4. Section 5 summarizes our study and points out some future research directions.

2 Preliminaries

Hereafter, we present some key notions which are of use to comprehend the remaining of the paper database [1,11,6].

Relation: Let $\mathcal{A} = \{a_1, \dots, a_m\}$ be a finite set of attributes. Each attribute a_i has a finite domain, denoted $\text{dom}(a_i)$, representing the values that a_i can take on. For a subset $X = \{a_i, \dots, a_j\}$ of \mathcal{A} , $\text{dom}(X)$ is the Cartesian product of the domains of the individual attributes in X . A relation r on \mathcal{A} is a finite set of tuples $\{t_1, \dots, t_n\}$ from \mathcal{A} to $\text{dom}(\mathcal{A})$ with the restriction that for each tuple $t \in r$, $t[X]$ must be in $\text{dom}(X)$, such that $X \subseteq \mathcal{A}$ and $t[X]$ denotes the restriction of the tuple t to X .

Functional dependency: Let r be a relation on \mathcal{A} . A functional dependency (FD) over \mathcal{A} is an expression $X \rightarrow A$ where $X \subseteq \mathcal{A}$ and $A \in \mathcal{A}$. We refer to X as the antecedent and A as the consequent. A FD $X \rightarrow A$ holds on r (denoted

$r \models X \rightarrow A$) if and only if $\forall(t_i, t_j) \in r, t_i[X] = t_j[X] \Rightarrow t_i[A] = t_j[A]$. A FD $X \rightarrow A$ is minimal if and only if $r \models X \rightarrow A$ and $\forall Z \subset X, r \not\models Z \rightarrow A$. We denote by \mathcal{F}_r the set of all functional dependencies satisfied by r .

Canonical cover: Let r be a relation on \mathcal{A} . The canonical cover of \mathcal{F}_r is defined as follows: $Cover(\mathcal{F}_r) = \{X \rightarrow A | X \subset \mathcal{A}, A \in \mathcal{A}, r \models X \rightarrow A, X \rightarrow A \text{ is minimal}\}$.

Agree sets of a relation: Let t and t' be two tuples of r and $X \subseteq \mathcal{A}$ be a set of attributes. Then, the tuples t and t' agree on X if and only if $t[X] = t'[X]$. Hence, according to t and t' , the agree set, denoted $Ag(t, t') = \{A | t[A] = t'[A], A \in \mathcal{A}\}$. $Ag(r) = \{Ag(t, t') | (t, t') \in r, t \neq t'\}$ denotes all agree sets of r .

Agree sets induced by a tuple: Let r be a relation and t be a tuple. Then, with respect to r , agree sets induced by the tuple t are $\{Ag(t, t') | t' \in r\}$ and denoted by $Ag(r)_t$.

Equivalence class: Let r be a relation on \mathcal{A} and t be a tuple. Then, for a given attribute $A \in \mathcal{A}$, the equivalence class of r with respect to t is $r(A)_t = \{t' | t[A] = t'[A], t' \in r\}$. We denote by $EC(r)_t = \{r(A)_t | A \in \mathcal{A}\}$ all equivalence classes of r with respect to t .

3 The MT-IncFDs Algorithm

We are at the beginning of the multi-core era. When combined with simultaneous multithreading technology, multi-core can provide powerful optimization opportunities, increasing system throughput substantially. Inspired by these modern technologies, we propose a multi-threaded algorithm, called MT-IncFDs, for incremental mining of functional dependencies, which harnesses multi-core systems. Let r be a relation on \mathcal{A} and t be a set of tuples to insert into r . The aim of MT-IncFDs consists to harness multi-core systems to determine the canonical cover $Cover(\mathcal{F}_{r'})$ of the updated relation $r' = r \cup t$ by taking advantage of the already discovered $Cover(\mathcal{F}_r)$. Since $r \subset r'$, then $\mathcal{F}_{r'} \subseteq \mathcal{F}_r$. In other words, there are two types of FDs:

- *the winners*: functional dependencies of \mathcal{F}_r which still hold on r' .
- *the losers*: functional dependencies of \mathcal{F}_r which do not hold on r'

For that, MT-IncFDs proceeds in two steps:

1. initializing $Cover(\mathcal{F}_{r'})$ with winners that belong to $Cover(\mathcal{F}_r)$;
2. maintaining losers of $Cover(\mathcal{F}_r)$ in order to deduce the new FDs of $Cover(\mathcal{F}_{r'})$.

3.1 Initialization of $Cover(\mathcal{F}_{r'})$ with Winners

MT-IncFDs assumes an alternate characterization of winners, which relies on the concept of agree sets induced by t .

Computing agree sets induced by t . In order to compute the agree sets induced by t , MT-INCFDs starts by computing the equivalence classes of r with respect to t . For that, we divide r among the p available threads. Each thread is given $|\mathcal{A}|/p$ equivalence classes to compute.

Example 1. Let us consider the relation of Table II and suppose that it contains only the six first tuples and suppose that we dispose of three threads $\{T_1, T_2, T_3\}$. In order to determine the equivalence classes of r with respect to t_7 , we assign to each thread $\frac{5}{3}$ equivalence classes. Hence, T_1 will compute $r(A)_{t_7}$ and $r(B)_{t_7}$. T_2 will compute $r(C)_{t_7}$ and $r(D)_{t_7}$. T_3 will compute $r(E)_{t_7}$. The obtained equivalence classes are :

$$\begin{aligned} r(A)_{t_7} &= \{t_1, t_3, t_4\}; \\ r(B)_{t_7} &= \{t_1\}; \\ r(C)_{t_7} &= \{t_5\}; \\ r(D)_{t_7} &= \{t_1, t_2, t_3, t_4\}; \\ r(E)_{t_7} &= \{t_1, t_2, t_4\}. \end{aligned}$$

Then, $EC(r)_{t_7} = \{r(A)_{t_7}, r(B)_{t_7}, r(C)_{t_7}, r(D)_{t_7}, r(E)_{t_7}\}$.

Table 1. Example of a relation

	A	B	C	D	E
t_1	1	100	1	2	50
t_2	4	101	1	2	50
t_3	1	102	2	2	70
t_4	1	200	1	2	50
t_5	2	101	3	3	100
t_6	2	200	1	3	70
t_7	1	100	3	2	50

It is important to mention that there is no dependence among the threads, because each thread has to compute equivalence classes of a fixed number of attributes. Furthermore, the relation r is accessed in read-only. Thus, concurrent accesses to r by multiple threads do not need synchronization and do not cause cache coherence invalidations and misses.

Afterwards, MT-INCFDs determines the maximal equivalence classes of r with respect to t denoted $MC(r)_t$. For that, each thread T_i checks the maximality of the equivalence classes at hand by comparing them to the equivalence classes computed by the remaining $p - 1$ threads. This test of maximality is done concurrently without need of synchronization since the equivalence classes are accessed in read-only.

Example 2. Let us continue with the previous example. T_1 has to check the maximality of $r(A)_{t_7}$ and $r(B)_{t_7}$ by comparing them to equivalence classes computed by T_2 and T_3 . T_2 has to check the maximality of $r(C)_{t_7}$ and $r(D)_{t_7}$ by comparing them to equivalence classes computed by T_1 and T_3 . T_3 has to check the maximality of $r(E)_{t_7}$ by comparing it to equivalence classes computed by T_1 and T_2 . The result of these test is $MC(r)_{t_7} = \{\{t_1, t_2, t_3, t_4\}, \{t_5\}\}$.

Next, MT-INCFDs uses Proposition 1 in order to compute the agree sets induced by t .

Proposition 1. *For a couple of tuples (t, t') , $Ag(t, t') = \{A | t' \in r(A)_t, r(A)_t \in EC(r)_t\}$. Hence, given a relation r and a tuple t , the agree sets induced by t with respect to r are given by: $Ag(r)_t = \{Ag(t, t') | t' \in c, c \in MC(r)_t\}$.*

For that, MT-INCFDs computes the number of couples (t, t') , which can be generated in order to find the average number of couples that ought to be generated by each thread. If S is this found average, MT-INCFDs goes over the sorted list of maximal equivalence classes by their respective sizes and assigns maximal classes consecutively for each thread until the cumulated number of couples is equal or greater than the average S . Once the collection of couples of tuples is generated, each thread should compute agree sets of the couples at hand. This step requires a scan of the equivalence classes. Since they are accessed in read-only, they can be thus shared without synchronization among all the threads.

Example 3. Let us continue with the previous example. First, maximal equivalence classes are sorted by their respective sizes. The number of couples of tuples that ought to be generated from maximal equivalence classes is equal to 6. Then the average $S = 2$. We start by assigning the only the first equivalence class to T_1 since the number of couples obtained from this equivalence class is greater than S . Hence, T_1 generates the following couples of tuples $\{(t_7, t_1), (t_7, t_2), (t_7, t_3), (t_7, t_4), (t_7, t_5)\}$. After, it deduces the agree sets induced by t_7 as follows.

$Ag(t_7, t_1) = ABDE$. As we note t_1 belongs to $r(A)_{t_7}, r(B)_{t_7}, r(D)_{t_7}$ and $r(E)_{t_7}$.

$Ag(t_7, t_2) = DE$. As we note t_2 belongs to $r(D)_{t_7}$ and $r(E)_{t_7}$

$Ag(t_7, t_3) = AD$. As we note t_3 belongs to $r(A)_{t_7}$ and $r(D)_{t_7}$.

$Ag(t_7, t_4) = ADE$. As we note t_4 belongs to $r(A)_{t_7}, r(D)_{t_7}$ and $r(E)_{t_7}$.

$Ag(t_7, t_5) = C$. As we note t_5 belongs to $r(C)_{t_7}$.

Consequently, $Ag(r)_{t_7} = \{ABDE, AD, ADE, DE, C\}$.

Identification of winners. MT-INCFDs uses Proposition 2 in order to identify the winners of $Cover(\mathcal{F}_r)$.

Proposition 2. *Let $Ag(r)_t$ be the agree sets induced by t with respect to r and $X \rightarrow A$ be a functional dependency of $Cover(\mathcal{F}_r)$. $X \rightarrow A$ is a winner if and only if $\exists Y \in Ag(r)_t$ such that $X \subseteq Y$ and $A \not\subseteq Y$.*

To achieve a suitable load-balancing, we divide $Cover(\mathcal{F}_r)$ among the p threads. Each thread is given $|Cover(\mathcal{F}_r)|/p$ FDs to check as explained by Proposition 2. It is important to mention that this step does not need synchronization since each thread can check the FDs independently.

Example 4. Let us consider the relation r of Table 1 and suppose that it contains only the five first tuples and we will insert the tuple t_6 . The following table gives $Cover(\mathcal{F}_r)$.

$BC \rightarrow A$	$AB \rightarrow C$	$A \rightarrow D$	$AB \rightarrow E$
$BD \rightarrow A$	$BD \rightarrow C$	$C \rightarrow D$	$BD \rightarrow E$
$BE \rightarrow A$	$E \rightarrow C$	$E \rightarrow D$	$C \rightarrow E$

After computing the agree sets induced by t_6 , we obtain $Ag(r)_{t_6} = \{C, E, BC, AD\}$. Since $Cover(\mathcal{F}_r)$ contains 12 FDs, then each thread has to check 4 FDs. Indeed, T_1 has to check $BC \rightarrow A$, $BD \rightarrow A$, $BE \rightarrow A$ and $AB \rightarrow C$. T_2 has to check $BD \rightarrow C$, $E \rightarrow C$, $A \rightarrow D$ and $C \rightarrow D$. T_3 has to check $E \rightarrow D$, $AB \rightarrow E$, $BD \rightarrow E$ and $C \rightarrow E$. According to Proposition 2, the threads deduce that $BD \rightarrow A$, $BE \rightarrow A$, $AB \rightarrow C$, $BD \rightarrow C$, $A \rightarrow D$, $AB \rightarrow E$ and $BD \rightarrow E$ are winner.

3.2 Maintaining Losers

Once winners are identified, we can easily determine the set of losers since losers are equal to $Cover(\mathcal{F}_r)$ -winners. For that, MT-INCFDs puts all losers in a queue and assigns them dynamically to a pool of threads on a "first come-first served" basis. In order to maintain a loser $X \rightarrow A$ at hand, each thread has to determine the agree sets induced by t which do not contain A . For that, each thread uses a tree recursive procedure whose recursion structure is based on depth-first search exploration of a search tree. Indeed we use a search tree having X as root and whose leaf nodes represent candidate antecedents of FDs which would replace $X \rightarrow A$. An arbitrary node, in level i , represents an antecedent obtained by considering maximal agree sets induced by t which do not contain A $\{G_1, G_2, \dots, G_i\}$. Each node represents a potential antecedent of a new FD that replace $X \rightarrow A$. In order to obtain a candidate of level $i + 1$, from a node Y of level i , we should consider the $i + 1^{th}$ maximal agree set induced by t . We distinguish two cases:

- if $Y \cap \overline{G}_{i+1} \neq \emptyset$, then this maximal agree set induced by t is ignored;
- else, we generate a child node equal to the union of Y and $\{e\}$, such that $\{e\} \in \overline{G}_{i+1}$.

For each leaf node Y , we have to verify that it does not exist a FD of $Cover(\mathcal{F}'_r)$ having an antecedent included in Y .

Once a thread has finished its task, it has to dequeue another unprocessed loser. Else, it polls the other running threads until it steals from a heavy loaded thread. For this step, there is no dependence among the threads, because each search tree corresponds to a disjoint set of potential antecedents of different FDs. Since each thread can proceed independently there is no synchronization while maintaining losers.

Example 5. Through this example we explain how a thread maintain a loser. Let us continue with the previous example. Let us maintain the loser $C \rightarrow E$. The agree sets induced by t_6 that do not contain E are $\{AD, BC\}$. We build a search tree having " C " as root. Then, we consider the first agree set AD . Since $C \cap \overline{AD} \neq \emptyset$, then this agree set is ignored. Afterwards, we consider the agree set BC . $C \cap \overline{BC} \neq \emptyset$. Consequently, we generate a child node AC that represents the

antecedent of the functional dependency $AC \rightarrow E$. Next, we generate a second child node CD that represents the antecedent of the functional dependency $CD \rightarrow E$. (We ignore $CE \rightarrow E$ because it is a trivial FD i.e., the consequent is contained in the antecedent).

4 Experimental Results

In this section, we assess the performances of MT-IncFDs and we highlight the benefit of using multi-core systems. Unfortunately, large multi-core cpus are still not available. For that, we conducted a set of experiments on a machine equipped with an Intel processor with 4 cores and 8 GB of RAM. We generated synthetic data sets (*i.e.*, relations) in order to control various parameters during the tests. We firstly create a table with $|\mathcal{A}|$ attributes in the database and then insert $|r|$ tuples one by one. Each inserted value depends on the parameter c , which is the rate of identical values. It controls the number of identical values in a column of the table. The experiments were carried out on: (1) a relation composed of 10000 tuples, 10 attributes and 50% of identical values per attributes and (2) a relation composed of 10000 tuples, 20 attributes and 50% of identical values per attributes. The curves of Figure 2 illustrates the execution time of MT-IncFDs

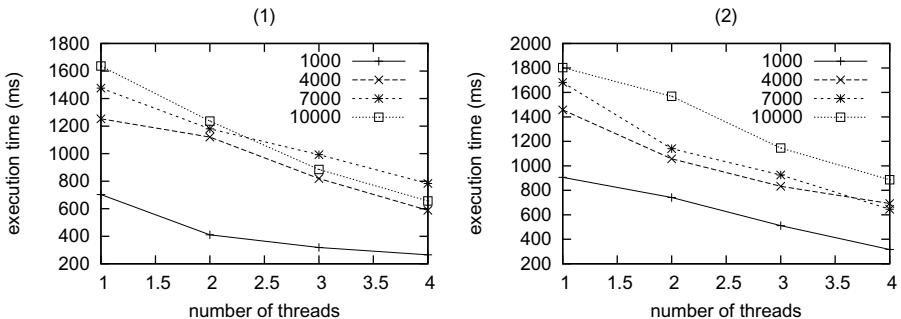


Fig. 1. Execution time of MT-IncFDs vs the variation of the number of threads

when, the 1000^{th} , 4000^{th} , 7000^{th} , 10000^{th} tuple is added. According to theses curves we can note that:

- the more we increase the number of threads the more execution time of MT-IncFDs decreases and we obtain a quasi-linear speed up;
- the proposed solution scales with both the increasing of number of attributes and the number of threads;
- the speed up of MT-IncFDs decreases slightly when we increase the number of tuples of the initial relation r . This is justified by the fact that the more the number of tuples increases, the more cache misses become important (for the step of computing of equivalence classes).

5 Conclusion

In this paper, we proposed the first parallel incremental algorithm for FD mining. Indeed, MT-INCFDs is a multi-threaded algorithm which harnesses the multi-core systems. We showed that our algorithm use data structures exhibiting high locality. Moreover, they are accessed in read-only. Thus, concurrent accesses to such data structures by multiple threads do not need synchronization and do not cause cache coherence invalidations and misses. The results of experiments carried out on synthetic databases showed a quasi-linear scale up with the number of cores for MT-INCFDs algorithm.

As perspective, it would be interesting to study the use of another parallel framework: clusters including clusters of multi-core machines in order to benefit from both architectures.

References

1. Abiteboul, S., Hull, R., Vianu, V.: Foundations of Databases. Addison-Wesley, Reading (1995)
2. Baixeries, J.: A formal concept analysis framework to mine functional dependencies. In: Proceeding of the Workshop on Mathematical Methods for Learning, Villa Geno, Italy (2004)
3. Demetrovics, J., Libkin, L., Muchnik, I.B.: Functional dependencies in relational databases: A lattice point of view. *Discrete Applied Mathematics* 40, 155–185 (1992)
4. Flesca, S., Furfaro, F., Greco, S., Zumpano, E.: Repairing inconsistent xml data with functional dependencies. In: Rivero, L.C., Doorn, J.H., Ferraggine, V.E. (eds.) Encyclopedia of Database Technologies and Applications, pp. 542–547. Idea Group, USA (2005)
5. Huhtala, Y., Kärkkäinen, J., Porkka, P., Toivonen, H.: Tane: An efficient algorithm for discovering functional and approximate dependencies. *Computer Journal* 42(2), 100–111 (1999)
6. Lopes, S., Petit, J., Lakhal, L.: Efficient discovery of functional dependencies and armstrong relations. In: Zaniolo, C., Grust, T., Scholl, M.H., Lockemann, P.C. (eds.) EDBT 2000. LNCS, vol. 1777, pp. 350–364. Springer, Heidelberg (2000)
7. Lopes, S., Petit, J., Lakhal, L.: Functional and approximate dependency mining: database and FCA points of view. *J. Exp. Theor. Artif. Intell.* 14(2-3), 93–114 (2002)
8. Maier, D.: The Theory of Relational Databases. Computer Science Press, Rockville (1983)
9. Mannila, H., Räihä, K.: Design by example: An application of armstrong relations. *Journal of Computer and System Sciences* 33(2), 126–141 (1986)
10. Mannila, H., Räihä, K.: Dependency inference. In: Proceedings of 13th International Conference on Very Large Data Bases, Brighton, England, September 1-4, pp. 155–158 (1987)
11. Mannila, H., Räihä, K.: Algorithms for inferring functional dependencies from relations. *Data Knowl. Eng.* 12(1), 83–99 (1994)
12. Novelli, N., Cicchetti, R.: Fun: An efficient algorithm for mining functional and embedded dependencies. In: Van den Bussche, J., Vianu, V. (eds.) ICDT 2001. LNCS, vol. 1973, pp. 189–203. Springer, Heidelberg (2000)

13. Tan, H.B.K., Zhao, Y.: Automated elicitation of functional dependencies from source codes of database transactions. *Information & Software Technology* 46(2), 109–117 (2004)
14. Wyss, C.M., Giannella, C., Robertson, E.L.: Fastfds: A heuristic-driven, depth-first algorithm for mining functional dependencies from relation instances - extended abstract. In: Kambayashi, Y., Winiwarter, W., Arikawa, M. (eds.) DaWaK 2001. LNCS, vol. 2114, pp. 101–110. Springer, Heidelberg (2001)
15. Yao, H., Hamilton, H.J., Butz, C.J.: Fd_mine: Discovering functional dependencies in a database using equivalences. In: Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002), Maebashi City, Japan, December 9-12, pp. 729–732 (2002)

Paraconsistent Semantics for Description Logics: A Comparison

Norihiro Kamide

Waseda Institute for Advanced Study, Waseda University,
1-6-1 Nishi Waseda, Shinjuku-ku, Tokyo 169-8050, Japan
`drnkamide08@kpd.biglobe.ne.jp`

Abstract. It is shown that four existing paraconsistent semantics (i.e., four-valued semantics, quasi-classical semantics, single-interpretation semantics and dual-interpretation semantics) for description logics are essentially the same semantics. To show this, two generalized and extended new semantics are introduced, and an equivalence between them is proved.

1 Introduction

Description logics (DLs) [2] are known to be valuable for obtaining logical foundations of web ontology languages. Some useful DLs including a standard description logic \mathcal{ALC} [14] have been studied by many researchers. *Paraconsistent (or inconsistency-tolerant) description logics* (PDLs) [6,7,8,9,11,12,13,15,18,19] have been studied to cope with inconsistencies which may frequently occur in an open world.

Some recent developments of PDLs may be briefly summarized as follows. An *inconsistency-tolerant four-valued terminological logic* was originally introduced by Patel-Schneider [13], three *inconsistency-tolerant constructive DLs*, which are based on intuitionistic logic, were studied by Odintsov and Wansing [11,12], some *paraconsistent four-valued DLs* including $\mathcal{ALC}4$ were studied by Ma et al. [6,7], some *quasi-classical DLs* were developed and studied by Zhang et al. [18,19], a sequent calculus for reasoning in four-valued DLs was introduced by Straccia [15], and an application of four-valued DL to information retrieval was studied by Meghini et al. [8,9]. A PDL called $\mathcal{P}\mathcal{ALC}$ has recently been proposed by Kamide [3] based on the idea of Kaneiwa [5] for his multiple-interpretation DL \mathcal{ALC}_\sim^n .

The logic $\mathcal{ALC}4$ [6], which is based on *four-valued semantics*, has a good translation into \mathcal{ALC} [14], and using this translation, the satisfiability problem for $\mathcal{ALC}4$ is shown to be decidable. But, $\mathcal{ALC}4$ and its variations have no classical negation (or complement). As mentioned in [16], classical and paraconsistent negations are known to be both useful for some knowledge-based systems. The quasi-classical DLs in [18,19], which are based on *quasi-classical semantics*, have the classical negation. But, translations of the quasi-classical DLs into the corresponding standard DLs were not proposed. $\mathcal{P}\mathcal{ALC}$ [3], which

is based on *dual-interpretation semantics*, have both the merits of $\mathcal{ALC}4$ and the quasi-classical DLs, i.e., it has the translation and the classical negation. The semantics of \mathcal{PALC} is taken over from the dual-consequence Kripke-style semantics for *Nelson's paraconsistent four-valued logic N4 with strong negation* [110]. The constructive PDLs in [11] are based on *single-interpretation semantics*, which can be seen as a DL-version of the single-consequence Kripke-style semantics for N4 [4].

The following natural question arises: What is the relationship among the single-interpretation semantics of the constructive PDLs, the dual-interpretation semantics of \mathcal{PALC} , the four-valued semantics of $\mathcal{ALC}4$, and the quasi-classical semantics of the quasi-classical DLs? This paper gives an answer to this question: These paraconsistent semantics are essentially the same semantics.¹ More precisely, we show the following. A new PDL, called \mathcal{QALC} , is introduced based on a generalized quasi-classical semantics. It can be seen that the quasi-classical semantics and the four-valued semantics are special cases of the \mathcal{QALC} semantics. An equivalence between \mathcal{QALC} and (a slightly modified version of) \mathcal{PALC} is proved. A new PDL, called \mathcal{SALC} , is introduced based on a modified single-interpretation semantics. An equivalence between \mathcal{SALC} and (a slightly modified version of) \mathcal{PALC} is proved. These results mean that the existing applications and theoretical results (e.g., decidability, complexity, embeddability and completeness) can be shared in these paraconsistent semantics.

2 Existing Paraconsistent Semantics

2.1 \mathcal{PALC} Semantics

In the following, we present the logic \mathcal{PALC} . The \mathcal{PALC} -concepts are constructed from atomic concepts, roles, \sim (paraconsistent negation), \neg (classical negation or complement), \sqcap (intersection), \sqcup (union), $\forall R$ (universal concept quantification) and $\exists R$ (existential concept quantification). We use the letters A and A_i for atomic concepts, the letter R for roles, and the letters C and D for concepts.

Definition 1. Concepts C are defined by the following grammar:

$$C ::= A \mid \neg C \mid \sim C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C$$

Definition 2. A paraconsistent interpretation \mathcal{PI} is a structure $\langle \Delta^{\mathcal{PI}}, .^{\mathcal{I}^+}, .^{\mathcal{I}^-} \rangle$ where

¹ This paper does not give a “comprehensive” comparison, since the existing paraconsistent semantics have some different constructors (or logical connectives), i.e., it is difficult to compare the whole parts of these existing semantics. But, this paper gives an “essential” comparison with respect to the common part with the constructors \sim (paraconsistent negation), \sqcap (intersection), \sqcup (union), $\forall R$ (universal concept quantification) and $\exists R$ (existential concept quantification). To obtain such a comparison with some exact proofs, we need some small modifications of the existing paraconsistent semantics.

1. $\Delta^{\mathcal{PI}}$ is a non-empty set,
2. $\cdot^{\mathcal{I}^+}$ is an interpretation function which assigns to every atomic concept A a set $A^{\mathcal{I}^+} \subseteq \Delta^{\mathcal{PI}}$ and to every role R a binary relation $R^{\mathcal{I}^+} \subseteq \Delta^{\mathcal{PI}} \times \Delta^{\mathcal{PI}}$,
3. $\cdot^{\mathcal{I}^-}$ is an interpretation function which assigns to every atomic concept A a set $A^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{PI}}$ and to every role R a binary relation $R^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{PI}} \times \Delta^{\mathcal{PI}}$,
4. for any role R , $R^{\mathcal{I}^+} = R^{\mathcal{I}^-}$.

The interpretation functions are extended to concepts by the following inductive definitions:

1. $(\sim C)^{\mathcal{I}^+} := C^{\mathcal{I}^-}$,
2. $(\neg C)^{\mathcal{I}^+} := \Delta^{\mathcal{PI}} \setminus C^{\mathcal{I}^+}$,
3. $(C \sqcap D)^{\mathcal{I}^+} := C^{\mathcal{I}^+} \cap D^{\mathcal{I}^+}$,
4. $(C \sqcup D)^{\mathcal{I}^+} := C^{\mathcal{I}^+} \cup D^{\mathcal{I}^+}$,
5. $(\forall R.C)^{\mathcal{I}^+} := \{a \in \Delta^{\mathcal{PI}} \mid \forall b [(a, b) \in R^{\mathcal{I}^+} \Rightarrow b \in C^{\mathcal{I}^+}]\}$,
6. $(\exists R.C)^{\mathcal{I}^+} := \{a \in \Delta^{\mathcal{PI}} \mid \exists b [(a, b) \in R^{\mathcal{I}^+} \wedge b \in C^{\mathcal{I}^+}]\}$,
7. $(\sim C)^{\mathcal{I}^-} := C^{\mathcal{I}^+}$,
8. $(\neg C)^{\mathcal{I}^-} := \Delta^{\mathcal{PI}} \setminus C^{\mathcal{I}^-}$,
9. $(C \sqcap D)^{\mathcal{I}^-} := C^{\mathcal{I}^-} \cup D^{\mathcal{I}^-}$,
10. $(C \sqcup D)^{\mathcal{I}^-} := C^{\mathcal{I}^-} \cap D^{\mathcal{I}^-}$,
11. $(\forall R.C)^{\mathcal{I}^-} := \{a \in \Delta^{\mathcal{PI}} \mid \exists b [(a, b) \in R^{\mathcal{I}^-} \wedge b \in C^{\mathcal{I}^-}]\}$,
12. $(\exists R.C)^{\mathcal{I}^-} := \{a \in \Delta^{\mathcal{PI}} \mid \forall b [(a, b) \in R^{\mathcal{I}^-} \Rightarrow b \in C^{\mathcal{I}^-}]\}$.

An expression $\mathcal{I}^* \models C$ ($*$ $\in \{+, -\}$) is defined as $C^{\mathcal{I}^*} \neq \emptyset$. A paraconsistent interpretation $\mathcal{PI} := \langle \Delta^{\mathcal{PI}}, \cdot^{\mathcal{I}^+}, \cdot^{\mathcal{I}^-} \rangle$ is a model of a concept C (denoted as $\mathcal{PI} \models C$) if $\mathcal{I}^* \models C$ ($*$ $\in \{+, -\}$). A concept C is said to be satisfiable in \mathcal{PALC} if there exists a paraconsistent interpretation \mathcal{PI} such that $\mathcal{PI} \models C$.

2.2 Four-Valued and Quasi-Classical Semantics

Some four-valued semantics in [6] were based on \mathcal{SHIQ} , $\mathcal{EL}++$, DL-Lite, etc., and the quasi-classical semantics in [19] was based on \mathcal{SHIQ} . The four-valued semantics in [6] has no classical negation, but has some new inclusion constructors such as strong inclusion. In addition, the quasi-classical semantics in [19] has two kinds of definitions called *QC weak semantics* and *QC strong semantics*. The following explanation is based on \mathcal{ALC} and QC weak semantics. We use the common language based on $\sim, \sqcap, \sqcup, \forall R, \exists R$ and/or \neg .²

The following definition is a slight modification of the definition of $\mathcal{ALC4}$ [6].

² We cannot compare the existing paraconsistent semantics (i.e., the four-valued semantics, the quasi-classical semantics, the single-interpretation semantics and the dual-interpretation semantics) themselves since the underlying DLs are different. Moreover, the motivations of introducing the existing semantics are completely different. For example, in the quasi-classical semantics, the main motivation is to satisfy three important inference rules: modus ponens, modus tollens and disjunctive syllogism. These inference rules are strongly dependent on a specific inclusion constructor \sqsubseteq and a specific QC entailment \models_Q . Thus, our comparison without \sqsubseteq is regarded as not so comprehensive or essential in the sense of the original motivation of the quasi-classical semantics.

Definition 3 (Four-valued semantics). A four-valued interpretation $\mathcal{I} := (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is defined using a pair $\langle P, N \rangle$ of subsets of $\Delta^{\mathcal{I}}$ and the projection functions $\text{proj}^+(\langle P, N \rangle) := P$ and $\text{proj}^-(\langle P, N \rangle) := N$. The interpretations are then defined as follows:³

1. a role R is assigned to a relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$,
2. for an atomic concept A , $A^{\mathcal{I}} := \langle P, N \rangle$ where $P, N \subseteq \Delta^{\mathcal{I}}$,
3. $(\sim C)^{\mathcal{I}} := \langle N, P \rangle$ if $C^{\mathcal{I}} = \langle P, N \rangle$,
4. $(C_1 \sqcap C_2)^{\mathcal{I}} := \langle P_1 \cap P_2, N_1 \cup N_2 \rangle$ if $C_i^{\mathcal{I}} = \langle P_i, N_i \rangle$ for $i = 1, 2$,
5. $(C_1 \sqcup C_2)^{\mathcal{I}} := \langle P_1 \cup P_2, N_1 \cap N_2 \rangle$ if $C_i^{\mathcal{I}} = \langle P_i, N_i \rangle$ for $i = 1, 2$,
6. $(\forall R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{I}} \mid \forall b [(a, b) \in R^{\mathcal{I}} \Rightarrow b \in \text{proj}^+(C^{\mathcal{I}})]\}, \{a \in \Delta^{\mathcal{I}} \mid \exists b [(a, b) \in R^{\mathcal{I}} \wedge b \in \text{proj}^-(C^{\mathcal{I}})]\} \rangle$,
7. $(\exists R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{I}} \mid \exists b [(a, b) \in R^{\mathcal{I}} \wedge b \in \text{proj}^+(C^{\mathcal{I}})]\}, \{a \in \Delta^{\mathcal{I}} \mid \forall b [(a, b) \in R^{\mathcal{I}} \Rightarrow b \in \text{proj}^-(C^{\mathcal{I}})]\} \rangle$.

The following definition is a slight modification of the definition of quasi-classical description logics [18,19].

Definition 4 (Quasi-classical semantics). A quasi-classical weak interpretation $\mathcal{I} := (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is defined using a pair $\langle +C, -C \rangle$ of subsets of $\Delta^{\mathcal{I}}$ without using projection functions. The interpretations are then defined as follows:⁴

1. a role R is assigned to a pair $R^{\mathcal{I}} = \langle +R, -R \rangle$ of binary relations $+R, -R \subseteq \Delta^{\mathcal{Q}\mathcal{I}} \times \Delta^{\mathcal{Q}\mathcal{I}}$,
2. for an atomic concept A , $A^{\mathcal{I}} := \langle +A, -A \rangle$ where $+A, -A \subseteq \Delta^{\mathcal{I}}$,
3. $(\sim C)^{\mathcal{I}} := \langle -C, +C \rangle$,
4. $(\neg C)^{\mathcal{I}} := \langle \Delta^{\mathcal{I}} \setminus +C, \Delta^{\mathcal{I}} \setminus -C \rangle$,
5. $(C_1 \sqcap C_2)^{\mathcal{I}} := \langle +C_1 \cap +C_2, -C_1 \cup -C_2 \rangle$,
6. $(C_1 \sqcup C_2)^{\mathcal{I}} := \langle +C_1 \cup +C_2, -C_1 \cap -C_2 \rangle$,
7. $(\forall R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{I}} \mid \forall b [(a, b) \in +R \Rightarrow b \in +C]\}, \{a \in \Delta^{\mathcal{I}} \mid \exists b [(a, b) \in -R \wedge b \in -C]\} \rangle$,
8. $(\exists R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{I}} \mid \exists b [(a, b) \in +R \wedge b \in +C]\}, \{a \in \Delta^{\mathcal{I}} \mid \forall b [(a, b) \in -R \Rightarrow b \in -C]\} \rangle$.

3 New Paraconsistent Semantics

3.1 \mathcal{QALC} Semantics

Similar notions and terminologies for \mathcal{PALC} are also used for the new logic \mathcal{QALC} . The \mathcal{QALC} -concepts are the same as the \mathcal{PALC} -concepts. The \mathcal{QALC} semantics is defined as a generalization and modification of the quasi-classical weak semantics defined in Definition 4. Thus, we use the term “quasi-classical” in the following definition.

³ In [6], the symbol \neg is used for the paraconsistent negation.

⁴ In [18,19], the symbols \neg and \sim are used for the paraconsistent negation and the classical negation, respectively.

Definition 5. A quasi-classical interpretation \mathcal{QI} is a structure $\langle \Delta^{\mathcal{QI}}, +, -, \cdot^{\mathcal{I}} \rangle$ where

1. $\Delta^{\mathcal{QI}}$ is a non-empty set,
2. $+(-)$ is a positive (negative, resp.) polarity function which assigns to every atomic concept A a set $+A \subseteq \Delta^{\mathcal{QI}}$ ($-A \subseteq \Delta^{\mathcal{QI}}$, resp.),
3. $\cdot^{\mathcal{I}}$ is an interpretation function which assigns to every atomic concept A a pair $A^{\mathcal{I}} = \langle +A, -A \rangle$ of sets $+A, -A \subseteq \Delta^{\mathcal{QI}}$ and to every role R a pair $R^{\mathcal{I}} = \langle +R, -R \rangle$ of binary relations $+R, -R \subseteq \Delta^{\mathcal{QI}} \times \Delta^{\mathcal{QI}}$,
4. for any role R , $+R = -R$.

The polarity functions are extended to concepts by the following inductive definitions:

1. $+(\sim C) := -C$,
2. $+(\neg C) := \Delta^{\mathcal{QI}} \setminus +C$,
3. $+(C \sqcap D) := +C \cap +D$,
4. $+(C \sqcup D) := +C \cup +D$,
5. $+(\forall R.C) := \{a \in \Delta^{\mathcal{QI}} \mid \forall b [(a, b) \in +R \Rightarrow b \in +C]\}$,
6. $+(\exists R.C) := \{a \in \Delta^{\mathcal{QI}} \mid \exists b [(a, b) \in +R \wedge b \in +C]\}$,
7. $-(\sim C) := +C$,
8. $-(\neg C) := \Delta^{\mathcal{QI}} \setminus -C$,
9. $-(C \sqcap D) := -C \cup -D$,
10. $-(C \sqcup D) := -C \cap -D$,
11. $-(\forall R.C) := \{a \in \Delta^{\mathcal{QI}} \mid \exists b [(a, b) \in -R \wedge b \in -C]\}$,
12. $-(\exists R.C) := \{a \in \Delta^{\mathcal{QI}} \mid \forall b [(a, b) \in -R \Rightarrow b \in -C]\}$.

The interpretation function is extended to concepts by:

$$C^{\mathcal{I}} := \langle +C, -C \rangle.$$

An expression $\mathcal{I} \models C$ is defined as $+C \neq \emptyset$ and $-C \neq \emptyset$. A quasi-classical interpretation $\mathcal{QI} := \langle \Delta^{\mathcal{QI}}, +, -, \cdot^{\mathcal{I}} \rangle$ is a model of a concept C (denoted as $\mathcal{QI} \models C$) if $\mathcal{I} \models C$. A concept C is said to be satisfiable in \mathcal{QALC} if there exists a quasi-classical interpretation \mathcal{QI} such that $\mathcal{QI} \models C$.

We have the following propositions, which mean that Definition 5 is essentially the same definitions as those of the original quasi-classical [18][19] and four-valued [6][7] semantics. See Definitions 4 and 3.

Proposition 6. Let $\cdot^{\mathcal{I}}$ be an interpretation function on a quasi-classical interpretation $\mathcal{QI} = \langle \Delta^{\mathcal{QI}}, +, -, \cdot^{\mathcal{I}} \rangle$. Then, the following conditions hold:

1. $(\sim C)^{\mathcal{I}} := \langle -C, +C \rangle$,
2. $(\neg C)^{\mathcal{I}} := \langle \Delta^{\mathcal{QI}} \setminus +C, \Delta^{\mathcal{QI}} \setminus -C \rangle$,
3. $(C \sqcap D)^{\mathcal{I}} := \langle +C \cap +D, -C \cup -D \rangle$,
4. $(C \sqcup D)^{\mathcal{I}} := \langle +C \cup +D, -C \cap -D \rangle$,
5. $(\forall R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{QI}} \mid \forall b [(a, b) \in +R \Rightarrow b \in +C]\}, \{a \in \Delta^{\mathcal{QI}} \mid \exists b [(a, b) \in -R \wedge b \in -C]\} \rangle$,

$$6. (\exists R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{Q}\mathcal{I}} \mid \exists b [(a, b) \in +R \wedge b \in +C]\}, \{a \in \Delta^{\mathcal{Q}\mathcal{I}} \mid \forall b [(a, b) \in -R \Rightarrow b \in -C]\} \rangle.$$

Proposition 7. Let $\cdot^{\mathcal{I}}$ be an interpretation function on a quasi-classical interpretation $\mathcal{Q}\mathcal{I} = \langle \Delta^{\mathcal{Q}\mathcal{I}}, +, -, \cdot^{\mathcal{I}} \rangle$. Let $+$ and $-$ be now represented by P and N, respectively. Also, $P(C)$ and $N(C)$ for a concept C be represented by P_C and N_C , respectively. Define $\text{proj}^+(P, N) := P$ and $\text{proj}^-(P, N) := N$. Then, the following conditions hold:

1. $(\sim C)^{\mathcal{I}} := \langle N_C, P_C \rangle$,
2. $(C \sqcap D)^{\mathcal{I}} := \langle P_C \cap P_D, N_C \cup N_D \rangle$,
3. $(C \sqcup D)^{\mathcal{I}} := \langle P_C \cup P_D, N_C \cap N_D \rangle$,
4. $(\forall R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{Q}\mathcal{I}} \mid \forall b [(a, b) \in \text{proj}^+(R^{\mathcal{I}}) \Rightarrow b \in \text{proj}^+(C^{\mathcal{I}})]\}, \{a \in \Delta^{\mathcal{Q}\mathcal{I}} \mid \exists b [(a, b) \in \text{proj}^-(R^{\mathcal{I}}) \wedge b \in \text{proj}^-(C^{\mathcal{I}})]\} \rangle$,
5. $(\exists R.C)^{\mathcal{I}} := \langle \{a \in \Delta^{\mathcal{Q}\mathcal{I}} \mid \exists b [(a, b) \in \text{proj}^+(R^{\mathcal{I}}) \wedge b \in \text{proj}^+(C^{\mathcal{I}})]\}, \{a \in \Delta^{\mathcal{Q}\mathcal{I}} \mid \forall b [(a, b) \in \text{proj}^-(R^{\mathcal{I}}) \Rightarrow b \in \text{proj}^-(C^{\mathcal{I}})]\} \rangle$.

Theorem 8 (Equivalence between QALC and PALC). For any concept C, C is satisfiable in QALC iff C is satisfiable in PALC.

Proof. • (\implies): Suppose that $\mathcal{Q}\mathcal{I} = \langle \Delta^{\mathcal{Q}\mathcal{I}}, +, -, \cdot^{\mathcal{I}} \rangle$ is a quasi-classical interpretation. Then, it is sufficient to construct a paraconsistent interpretation $\mathcal{P}\mathcal{I} = \langle \Delta^{\mathcal{P}\mathcal{I}}, \cdot^+, \cdot^- \rangle$ such that, for any concept C, $\mathcal{Q}\mathcal{I} \models C$ iff $\mathcal{P}\mathcal{I} \models C$. We define a paraconsistent interpretation $\mathcal{P}\mathcal{I}$ by:

1. $\Delta^{\mathcal{P}\mathcal{I}} := \Delta^{\mathcal{Q}\mathcal{I}}$,
2. \cdot^+ is an interpretation function which assigns to every atomic concept A a set $A^{\mathcal{I}^+} = +A \subseteq \Delta^{\mathcal{Q}\mathcal{I}}$ and to every role R a binary relation $R^{\mathcal{I}^+} = +R \subseteq \Delta^{\mathcal{Q}\mathcal{I}} \times \Delta^{\mathcal{Q}\mathcal{I}}$,
3. \cdot^- is an interpretation function which assigns to every atomic concept A a set $A^{\mathcal{I}^-} = -A \subseteq \Delta^{\mathcal{Q}\mathcal{I}}$ and to every role R a binary relation $R^{\mathcal{I}^-} = -R \subseteq \Delta^{\mathcal{Q}\mathcal{I}} \times \Delta^{\mathcal{Q}\mathcal{I}}$.

Then, we have the fact: for any role R, $R^{\mathcal{I}^+} = R^{\mathcal{I}^-}$.

It is sufficient to show the following claim which implies the required fact. For any concept C,

1. $+C = C^{\mathcal{I}^+}$,
2. $-C = C^{\mathcal{I}^-}$.

By (simultaneous) induction on C. We show some cases.

Case $C \equiv A$ (A is an atomic concept): For 1, we have the following by the definition: $+A = A^{\mathcal{I}^+}$. For 2, we have the following by the definition: $-A = A^{\mathcal{I}^-}$.

Case $C \equiv \sim D$: For 1, we have: $+(\sim D) = -D = D^{\mathcal{I}^-}$ (by induction hypothesis for 2) = $(\sim D)^{\mathcal{I}^+}$. For 2, we have: $-(\sim D) = +D = D^{\mathcal{I}^+}$ (by induction hypothesis for 1) = $(\sim D)^{\mathcal{I}^-}$.

Case $C \equiv \forall R.D$: For 1, we have:

$$+(\forall R.D)$$

$$\begin{aligned}
&= \{a \in \Delta^{\mathcal{QI}} \mid \forall b [(a, b) \in +R \Rightarrow b \in +D]\} \\
&= \{a \in \Delta^{\mathcal{PI}} \mid \forall b [(a, b) \in R^{\mathcal{I}^+} \Rightarrow b \in D^{\mathcal{I}^+}]\} \text{ (by induction hypothesis for 1)} \\
&= (\forall R.D)^{\mathcal{I}^+}.
\end{aligned}$$

For 2, we have:

$$\begin{aligned}
&\neg(\forall R.D) \\
&= \{a \in \Delta^{\mathcal{QI}} \mid \exists b [(a, b) \in -R \wedge b \in -D]\}, \\
&= \{a \in \Delta^{\mathcal{PI}} \mid \exists b [(a, b) \in R^{\mathcal{I}^-} \wedge b \in D^{\mathcal{I}^-}]\} \text{ (by induction hypothesis for 2),} \\
&= (\forall R.D)^{\mathcal{I}^-}.
\end{aligned}$$

• (\Leftarrow): Suppose that $\mathcal{PI} = \langle \Delta^{\mathcal{PI}}, \cdot^{\mathcal{I}^+}, \cdot^{\mathcal{I}^-} \rangle$ is a paraconsistent interpretation. Then, it is sufficient to construct a quasi-classical interpretation $\mathcal{QI} = \langle \Delta^{\mathcal{QI}}, +, -, \cdot^{\mathcal{I}} \rangle$ such that, for any concept C , $\mathcal{PI} \models C$ iff $\mathcal{QI} \models C$. We define a quasi-classical interpretation \mathcal{QI} by:

1. $\Delta^{\mathcal{QI}} := \Delta^{\mathcal{PI}}$,
2. $+(-)$ is a positive (negative, resp.) polarity function which assigns to every atomic concept A a set $+A = A^{\mathcal{I}^+} \subseteq \Delta^{\mathcal{PI}}$ ($-A = A^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{PI}}$, resp.),
3. $\cdot^{\mathcal{I}}$ is an interpretation function which assigns to every atomic concept A a pair $A^{\mathcal{I}} = \langle +A, -A \rangle$ of sets $+A = A^{\mathcal{I}^+}, -A = A^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{PI}}$ and to every role R a pair $R^{\mathcal{I}} = \langle +R, -R \rangle$ of binary relations $+R = R^{\mathcal{I}^+}, -R = R^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{PI}} \times \Delta^{\mathcal{PI}}$.

Then, we have the fact: for any role R , $+R = -R$.

It is sufficient to show the following claim which implies the required fact. For any concept C ,

1. $C^{\mathcal{I}^+} = +C$,
2. $C^{\mathcal{I}^-} = -C$.

Since this claim can be shown in the same way as in the claim of the direction (\Rightarrow), the proof is omitted here. ■

3.2 \mathcal{SALC} Semantics

We introduce a new logic \mathcal{SALC} , which has a single-interpretation function. The idea of this formulation is inspired from the paraconsistent semantics for a constructive PDL proposed in [11]. These single-interpretation semantics can also be adapted to Nelson's paraconsistent logic (see [4]).

Similar notions and terminologies for \mathcal{PALC} are also used for \mathcal{SALC} . The \mathcal{SALC} -concepts are the same as the \mathcal{PALC} -concepts.

Definition 9. Let Φ be the set of atomic concepts and Φ^\sim be the set $\{\sim A \mid A \in \Phi\}$. A single paraconsistent interpretation \mathcal{SI} is a structure $\langle \Delta^{\mathcal{ST}}, \cdot^{\mathcal{I}} \rangle$ where

1. $\Delta^{\mathcal{SI}}$ is a non-empty set,
2. $\cdot^{\mathcal{I}}$ is an interpretation function which assigns to every atomic (or negated atomic) concept $A \in \Phi \cup \Phi^\sim$ a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{SI}}$ and to every role R a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{SI}} \times \Delta^{\mathcal{SI}}$.

The interpretation function is extended to concepts by the following inductive definitions:

1. $(\neg C)^{\mathcal{I}} := \Delta^{\mathcal{SI}} \setminus C^{\mathcal{I}}$,
2. $(C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}$,
3. $(C \sqcup D)^{\mathcal{I}} := C^{\mathcal{I}} \cup D^{\mathcal{I}}$,
4. $(\forall R.C)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{SI}} \mid \forall b [(a, b) \in R^{\mathcal{I}} \Rightarrow b \in C^{\mathcal{I}}]\}$,
5. $(\exists R.C)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{SI}} \mid \exists b [(a, b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}]\}$,
6. $(\sim\sim C)^{\mathcal{I}} := C^{\mathcal{I}}$,
7. $(\sim\neg C)^{\mathcal{I}} := \Delta^{\mathcal{SI}} \setminus (\sim C)^{\mathcal{I}}$,
8. $(\sim(C \sqcap D))^{\mathcal{I}} := (\sim C)^{\mathcal{I}} \cup (\sim D)^{\mathcal{I}}$,
9. $(\sim(C \sqcup D))^{\mathcal{I}} := (\sim C)^{\mathcal{I}} \cap (\sim D)^{\mathcal{I}}$,
10. $(\sim\forall R.C)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{SI}} \mid \exists b [(a, b) \in R^{\mathcal{I}} \wedge b \in (\sim C)^{\mathcal{I}}]\}$,
11. $(\sim\exists R.C)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{SI}} \mid \forall b [(a, b) \in R^{\mathcal{I}} \Rightarrow b \in (\sim C)^{\mathcal{I}}]\}$.

An expression $\mathcal{I} \models C$ is defined as $C^{\mathcal{I}} \neq \emptyset$. A single paraconsistent interpretation $\mathcal{SI} := \langle \Delta^{\mathcal{SI}}, \cdot^{\mathcal{I}} \rangle$ is a model of a concept C (denoted as $\mathcal{SI} \models C$) if $\mathcal{I} \models C$. A concept C is said to be satisfiable in \mathcal{SALC} if there exists a single paraconsistent interpretation \mathcal{SI} such that $\mathcal{SI} \models C$.

Theorem 10 (Equivalence between \mathcal{SALC} and \mathcal{PACL}). For any concept C , C is satisfiable in \mathcal{SALC} iff C is satisfiable in \mathcal{PACL} .

Proof. Let Φ be the set of atomic concepts, Φ^\sim be the set $\{\sim A \mid A \in \Phi\}$, and Π be the set of roles.

• (\implies): Suppose that $\mathcal{SI} = \langle \Delta^{\mathcal{SI}}, \cdot^{\mathcal{I}} \rangle$ is a single paraconsistent interpretation such that $\cdot^{\mathcal{I}}$ has the domain $\Phi \cup \Phi^\sim \cup \Pi$. Then, it is sufficient to construct a paraconsistent interpretation $\mathcal{PI} = \langle \Delta^{\mathcal{PI}}, \cdot^{\mathcal{I}^+}, \cdot^{\mathcal{I}^-} \rangle$ such that, for any concept C , $\mathcal{SI} \models C$ iff $\mathcal{PI} \models C$. We define a paraconsistent interpretation \mathcal{PI} by:

1. $\Delta^{\mathcal{PI}} := \Delta^{\mathcal{SI}}$,
2. $\cdot^{\mathcal{I}^+}$ is an interpretation function which assigns to every atomic concept $A \in \Phi$ a set $A^{\mathcal{I}^+} \subseteq \Delta^{\mathcal{SI}}$ and to every role R a binary relation $R^{\mathcal{I}^+} \subseteq \Delta^{\mathcal{SI}} \times \Delta^{\mathcal{SI}}$,
3. $\cdot^{\mathcal{I}^-}$ is an interpretation function which assigns to every atomic concept $A \in \Phi$ a set $A^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{SI}}$ and to every role R a binary relation $R^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{SI}} \times \Delta^{\mathcal{SI}}$,
4. for any role R , $R^{\mathcal{I}^+} = R^{\mathcal{I}^-} = R^{\mathcal{I}}$,
5. the following conditions hold:
 - (a) $A^{\mathcal{I}^+} = A^{\mathcal{I}}$,
 - (b) $A^{\mathcal{I}^-} = (\sim A)^{\mathcal{I}}$.

It is noted that $\cdot^{\mathcal{I}^+}$ and $\cdot^{\mathcal{I}^-}$ have the domain $\Phi \cup \Pi$.

It is sufficient to show the following claim which implies the required fact. For any concept C ,

1. $C^{\mathcal{I}} = C^{\mathcal{I}^+}$,
2. $(\sim C)^{\mathcal{I}} = C^{\mathcal{I}^-}$.

By (simultaneous) induction on C . We show only the following case.

Case $C \equiv \sim D$: For 1, we have: $(\sim D)^{\mathcal{I}} = D^{\mathcal{I}^-}$ (by induction hypothesis for 2) $= (\sim D)^{\mathcal{I}^+}$. For 2, we have: $(\sim \sim D)^{\mathcal{I}} = D^{\mathcal{I}} = D^{\mathcal{I}^+}$ (by induction hypothesis for 1) $= (\sim D)^{\mathcal{I}^-}$.

• (\Leftarrow): Suppose that $\mathcal{PI} = \langle \Delta^{\mathcal{PI}}, .^{\mathcal{I}^+}, .^{\mathcal{I}^-} \rangle$ is a paraconsistent interpretation such that $.^{\mathcal{I}^+}$ and $.^{\mathcal{I}^-}$ have the domain $\Phi \cup \Pi$. Then, it is sufficient to construct a single paraconsistent interpretation $\mathcal{SI} = \langle \Delta^{\mathcal{SI}}, .^{\mathcal{I}} \rangle$ such that, for any concept C , $\mathcal{PI} \models C$ iff $\mathcal{SI} \models C$. We define a single paraconsistent interpretation \mathcal{SI} by:

1. $\Delta^{\mathcal{SI}} := \Delta^{\mathcal{PI}}$,
2. $.^{\mathcal{I}}$ is an interpretation function which assigns to every atomic (or negated atomic) concept $A \in \Phi \cup \Phi^\sim$ a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{PI}}$ and to every role R a binary relation $R^{\mathcal{I}} = R^{\mathcal{I}^+} = R^{\mathcal{I}^-} \subseteq \Delta^{\mathcal{PI}} \times \Delta^{\mathcal{PI}}$,
3. the following conditions hold:
 - (a) $A^{\mathcal{I}} = A^{\mathcal{I}^+}$,
 - (b) $(\sim A)^{\mathcal{I}} = A^{\mathcal{I}^-}$.

It is noted that $.^{\mathcal{I}}$ has the domain $\Phi \cup \Phi^\sim \cup \Pi$.

It is sufficient to show the following claim which implies the required fact. For any concept C ,

1. $C^{\mathcal{I}^+} = C^{\mathcal{I}}$,
2. $C^{\mathcal{I}^-} = (\sim C)^{\mathcal{I}}$.

Since this claim can be shown in the same way as in the claim of the direction (\Rightarrow), the proof is omitted here. ■

4 Conclusions

In this paper, new paraconsistent description logics \mathcal{QALC} and \mathcal{SALC} were introduced, and the equivalence among \mathcal{QALC} , \mathcal{SALC} and \mathcal{PALC} were proved. The \mathcal{QALC} -semantics is regarded as a generalization of both the four-valued semantics [6][7] and the quasi-classical semantics [18][19]. The \mathcal{SALC} -semantics is regarded as a small modification of the single-interpretation semantics [11][12]. The \mathcal{PALC} -semantics [3], also called dual-interpretation semantics, was taken over from the dual-consequence Kripke-style semantics for Nelson's paraconsistent logic N4 [1][10].

Acknowledgments. This research was partially supported by the Japanese Ministry of Education, Culture, Sports, Science and Technology, Grant-in-Aid for Young Scientists (B) 20700015.

References

1. Almukdad, A., Nelson, D.: Constructible falsity and inexact predicates. *Journal of Symbolic Logic* 49, 231–233 (1984)
2. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.F. (eds.): *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, Cambridge (2003)
3. Kamide, N.: Paraconsistent description logics revisited. In: Proceedings of the 23rd International Workshop on Description Logics (DL 2010), CEUR Workshop Proceedings, vol. 573 (2010)
4. Kamide, N.: An embedding-based completeness proof for Nelson's paraconsistent logic. *Bulletin of the Section of Logic* 39(3/4), 205–214 (2010)
5. Kaneiwa, K.: Description logics with contraries, contradictories, and subcontraries. *New Generation Computing* 25(4), 443–468 (2007)
6. Ma, Y., Hitzler, P., Lin, Z.: Algorithms for paraconsistent reasoning with OWL. In: Franconi, E., Kifer, M., May, W. (eds.) *ESWC 2007. LNCS*, vol. 4519, pp. 399–413. Springer, Heidelberg (2007)
7. Ma, Y., Hitzler, P., Lin, Z.: Paraconsistent reasoning for expressive and tractable description logics. In: Proceedings of the 21st International Workshop on Description Logic (DL 2008), CEUR Workshop Proceedings, vol. 353 (2008)
8. Meghini, C., Straccia, U.: A relevance terminological logic for information retrieval. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 197–205 (1996)
9. Meghini, C., Sebastiani, F., Straccia, U.: Mirlog: A logic for multimedia information retrieval. In: *Uncertainty and Logics: Advanced Models for the Representation and Retrieval of Information*, pp. 151–185. Kluwer Academic Publishing, Dordrecht (1998)
10. Nelson, D.: Constructible falsity. *Journal of Symbolic Logic* 14, 16–26 (1949)
11. Odintsov, S.P., Wansing, H.: Inconsistency-tolerant description logic: Motivation and basic systems. In: Hendricks, V.F., Malinowski, J. (eds.) *Trends in Logic: 50 Years of Studia Logica*, pp. 301–335. Kluwer Academic Publishers, Dordrecht (2003)
12. Odintsov, S.P., Wansing, H.: Inconsistency-tolerant Description Logic. Part II: Tableau Algorithms. *Journal of Applied Logic* 6, 343–360 (2008)
13. Patel-Schneider, P.F.: A four-valued semantics for terminological logics. *Artificial Intelligence* 38, 319–351 (1989)
14. Schmidt-Schauss, M., Smolka, G.: Attributive concept descriptions with complements. *Artificial Intelligence* 48, 1–26 (1991)
15. Straccia, U.: A sequent calculus for reasoning in four-valued description logics. In: Galmiche, D. (ed.) *TABLEAUX 1997. LNCS*, vol. 1227, pp. 343–357. Springer, Heidelberg (1997)
16. Wagner, G.: A database needs two kinds of negations. In: Thalheim, B., Gerhardt, H.-D., Demetrovics, J. (eds.) *MFDBS 1991. LNCS*, vol. 495, pp. 357–371. Springer, Heidelberg (1991)
17. Wansing, H.: The Logic of Information Structures. *LNCS (LNAI)*, vol. 681, pp. 1–163. Springer, Heidelberg (1993)
18. Zhang, X., Lin, Z.: Paraconsistent reasoning with quasi-classical semantic in \mathcal{ALC} . In: Calvanese, D., Lausen, G. (eds.) *RR 2008. LNCS*, vol. 5341, pp. 222–229. Springer, Heidelberg (2008)
19. Zhang, X., Qi, G., Ma, Y., Lin, Z.: Quasi-classical semantics for expressive description logics. In: Proceedings of the 22nd International Workshop on Description Logic (DL 2009), CEUR Workshop Proceedings, vol. 477 (2009)

Author Index

- Abbasi, Alireza II-256
Abbattista, Fabio I-249
Abe, Akinori II-495
Abe, Keiichi III-106
Adachi, Tomoya III-11
Adachi, Yoshinori IV-80, IV-117
Adrian, Benjamin II-420
Ahmadinia, Ali III-453, III-462, III-472
Aimi, Annuar H.B.M. III-415
Akdoğan, Erhan I-271
al Agroudy, Passant II-410
Alamgir Hossain, M. I-151
Albert, Dietrich IV-261
Alghowinem, Sharifa I-377
Alizadeh, Hosein I-21
Ambiah, Norbaitiah III-346
Anderson, Terry I-161
Aoki, Kumiko III-548
Aoki, Shingo IV-242
Aoki, Yuki III-558
Arakawa, Yutaka IV-252
Arasawa, Ryosuke IV-14
Arghir, Stefan I-1, I-72
Argotte, Liliana II-94
Aritsugi, Masayoshi IV-53
Arotaritei, Dragos I-1, I-72
Arroyo-Figueroa, G. II-94
Atteya, Walid Adly I-151
Aude, Aufaure Marie II-41
Aufaure, Marie-Aude II-538
Azzeh, Mohammad II-315
- Baba, A. Fevzi I-90
Baba, Norio II-366
Badaracco, Miguel II-124
Bae, Junghyo I-289
Banba, Hideaki III-227
Bao, Yongguang IV-98
Baralis, Elena II-515
Bardis, Georgios I-347
Bardone, Emanuele II-486
Batres, Rafael III-395
Batsakis, Sotiris I-558
Baumann, Stephan I-495
- Beigi, Akram I-110
Belanche, Lluís I-100
Beloe, Neil III-483
Ben-Abdallah, Hanene I-407
Ben Ahmed, Mohamed I-516
Benites, Fernando I-579
Ben Romdhane, Nadra I-407
Berns, Karsten IV-167
Bi, Yaxin I-161
Bianchi, Alessandro I-249
Biernacki, Paweł I-418
Boland, Katarina IV-366
Bonachela, Patricia II-611
Bondarenko, Andrey I-62
Boongasame, Laor I-230
Borzemski, Leszek II-581
Bouamama, Sadok II-325
Bouki, Yoshihiko III-153
Bravo, Jose II-296
Breiner, Kai IV-136
Brezovan, Marius I-505
Bridge, David III-483
Brucker, Florian I-579
Brusey, James III-483
Bueno, Gloria II-611
Bui, Len I-436
Burdescu, Dumitru Dan I-505
Burgin, Mark II-524
Burns, Nicola I-161
- Cairó, Osvaldo I-316, II-306
Cálad-Álvarez, Alejandro II-601
Carlson, Christoph I-239
Carrasco, Eduardo II-611
Castellano, G. II-84
Ceccarelli, Michele I-568
Ceci, Michelangelo II-559
Cerquitelli, Tania II-515
Chang, Wei-Lun II-285
Chen, Bo-Tsuen II-382
Chen, Chia-Chen II-382
Chen, Hanxiong II-21
Chen, Mu-Yen II-382
Chen, Weiqin I-239, III-558

- Chetty, Girija I-436
 Chiang, Hsiu-Sen II-382
 Chiusano, Silvia II-515
 Chookaew, Sasithorn II-134
 Chowdhury, Nihad K. II-355
 Chu, Yuyi III-237
 Constantin, George I-72
 Cox, Robert I-377
 Coyne, Bob IV-378
 Csipkes, D. III-472
 Csipkes, G. III-472
 Cuzzocrea, Alfredo II-559, II-571
- Dahal, Keshav I-151
 Davies, Gwyn III-433, IV-425
 Decker, Hendrik II-548
 De Felice, Fabio I-249
 Dengel, Andreas I-397, I-495, IV-187, IV-212, IV-222
 de Schryver, Christian IV-177
 Deßloch, Stefan IV-126
 de Vey Mestdagh, Kees (C.N.J.) II-524
 di Bella, Enrico II-392
 Dolog, Peter II-505
 Doran, Rodica-Elena II-265
- Ebert, Sebastian IV-222
 Eichhoff, Julian R. I-387
 Eklund, Tomas II-186
 Endo, Yasunori I-131
 Enomoto, Yoshitaro III-246
 Eto, Kaoru III-31
- Fanelli, A.M. II-84
 Faragó, Paul IV-455
 Farjami, Sahar III-499
 Farkas, Ioana-Iuliana II-265
 Feng, Yaokai IV-195
 Fernandez-Canque, Hernando III-453, III-462, III-472
 Feştilă, Lelia IV-455
 Firmansyah, Tatan III-395
 Fontechá, Jesús II-296
 Fouladgar, Hani II-214
 Frank, Anette IV-366
 Fruhata, Takashi III-11
 Fuchino, Tetsuo III-423
 Fujii, Satoru III-86, III-144
 Fujita, Tomoki III-77
 Fujita, Yoshikatsu III-378
- Fujiwara, Minoru IV-288
 Fujiwara, Reiko II-447
 Fukuda, Akira IV-252
 Fukui, Shinji IV-108
 Fukumura, Yoshimi III-499, III-548
 Fukushima, Taku II-31
 Furuhata, Takashi III-1
 Furuse, Kazutaka II-21
 Futamura, Kazuya IV-117
- Gaillardet, Jean-Marie IV-126
 Gälätuş, Ramona III-493
 Garnik, Igor II-657
 Gasmi, Ghada I-590
 Gaura, Elena III-483
 Gavrilova, Tatiana A. I-337
 Geibel, Peter I-538
 Genquan, Ren I-528
 Georgieva, Olga I-82
 Gesell, Manuel IV-167
 Ghezala, Henda Ben II-538
 Gill, Balpreet II-440
 Goda, Kazumasa II-154
 Godehardt, Eicke II-402
 Golfinopoulos, Vassilios I-347
 Gotoda, Naka III-21, III-520
 Graña, Manuel II-611
 Grand, Alberto II-515
 Grauer, Manfred III-56
 Grimaudo, Luigi II-515
 Grivas, Stella Gatzia II-51, II-275
 Grosvenor, Roger III-433, IV-425
 Grundmann, Thomas IV-126
 Grzech, Adam II-687
 Guardati, Silvia I-316
 Guijarro, Frank II-611
- Ha, Taehyun I-289
 Hajer, Baazaoui II-41
 Håkansson, Anne IV-415
 Hamaguchi, Takashi III-415
 Hamasuna, Yukihiro I-131
 Hammami, Mohamed I-407
 Han, Lee Chen II-366
 Hanabusa, Hisatomo III-596
 Hanaue, Koichi IV-14
 Hangos, Katalin M. III-385
 Hara, Chihiro IV-288
 Harada, Kouji IV-308
 Haraguchi, Makoto II-457

- Hartung, Ronald L. IV-409
 Hasegawa, Mikio III-587
 Hasegawa, Shinobu I-484
 Hashimoto, Kiyota IV-261
 Hashimoto, Takako II-73
 Hashizume, Ayako III-197
 Hattori, Fumio IV-44
 Hattori, Masanori III-67
 Hayashi, Yuki II-104, III-578, III-637
 Heckemann, Karl IV-167
 Helard, Maryline III-116
 Hellevang, Mathias I-239
 Henda, Ben Ghezala II-41
 Henmi, Fumiaki III-227
 Hernandez, Yasmin II-94
 Hervás, Ramón II-296
 Hintea, Diana III-483
 Hintea, Sorin III-443, III-472, IV-455
 Hirokawa, Sachio II-457
 Hirosue, Noriaki III-177
 Hochin, Teruhisa IV-1
 Homenda, Wladyslaw IV-232
 Horiguchi, Ryota III-596
 Horiuchi, Kensuke III-67
 Hossain, Liaquat II-256
 Huang, Xu I-436
 Hudelot, Céline I-538
 Hung, Tzu-Shiang II-285
- Ichikawa, Teruhisa III-134
 Igarashi, Harukazu I-120
 Iida, Takayuki III-67
 Iijima, Chie III-246
 Iizuka, Kayo III-366
 Iizuka, Yasuki III-366
 Ikeda, Mitsuru IV-288
 Imono, Misako I-367, I-474
 Inoue, Etsuko III-153
 Inoue, Shuki IV-242
 Inuzuka, Nobuhiro IV-89
 Ionescu, Florin I-1, I-72
 Iribe, Yurie III-548
 Ishida, Yoshiteru IV-308, IV-318,
 IV-328, IV-338, IV-348, IV-357
 Ishihara, Seiji I-120
 Ishii, Naohiro III-616, IV-73, IV-98
 Iswandy, Kuncup II-335, IV-155
 Ito, Hideaki IV-34
 Ito, Nobuhiro III-616
 Itou, Junko III-77, III-126
- Ivanciu, Laura III-443
 Iwahori, Yuji IV-80, IV-108, IV-117
 Iwamura, Masakazu IV-204
 Iwashita, Motoi III-256, III-275
 Iwata, Kazunori III-616
- Jabban, Ahmad III-116
 Jevtic, Dragan I-357
 Jianmin, Wang I-528
 Jimbo, Takashi IV-73
 Jin, Ping II-505
 Jlaiel, Nahla I-516
 Johansen, Bjarte III-558
 Jones, Leslie I-377
 Jumutc, Vilen I-62
 Jung, Matthias IV-177
 Juszczyszyn, Krzysztof II-687
- Kahl, Gerrit IV-187
 Kambayashi, Yasushi I-260, I-280
 Kameda, Hisashi III-606
 Kamide, Norihiro I-599, II-225, II-235,
 II-246
 Kamińska-Chuchmała, Anna II-581
 Kanematsu, Hideyuki III-499
 Kanenishi, Kazuhide III-520
 Karadgi, Sachin III-56
 Kashihara, Akihiro I-484, II-165
 Kataoka, Nobuhiro III-207
 Katayama, Shigetomo I-280
 Katsumata, Yuji IV-328
 Kawaguchi, Masashi IV-73
 Kawai, Atsuo II-144
 Kawai, Hideki II-63
 Kawano, Kouji III-423
 Kholod, Marina III-304
 Kikuchi, Masaaki III-67
 Kim, Daekyeong I-289
 Kim, Ikno III-237
 Kim, Jinseog II-203
 Kimura, Naoki III-415
 Kise, Koichi I-397, IV-204, IV-212
 Kitagawa, Hiroyuki II-21
 Kitajima, Teiji III-423
 Kitami, Kodai III-285
 Kitamura, Akira II-447
 Kitani, Tomoya III-134
 Kitasuka, Teruaki IV-53
 Klawonn, Frank I-82
 Klein, Andreas IV-146

- Klinkigt, Martin I-397, IV-212
 Kohtsuka, Takafumi I-280
 Kojima, Masanori III-207
 Kojiri, Tomoko II-104, III-578, III-637
 Koketsu, Hiroaki III-616
 König, Andreas I-424, II-335, IV-155
 Köppen, Mario III-177
 Korn, Ralf IV-177
 Koschel, Arne II-275
 Koshimizu, Hiroyasu IV-34
 Kostiuk, Anton IV-177
 Kotulski, Leszek I-180, I-190
 Kouno, Shouji III-275
 Kowalczyk, Ryszard I-200
 Krömker, Susanne IV-366
 Kubo, Masao III-627
 Kuboyama, Tetsuji II-73
 Kucharski, Bartosz II-640
 Kunieda, Kazuo II-63
 Kunimune, Hisayoshi III-529
 Kurahashi, Setsuya III-356
 Kuroda, Chiaki III-405
 Kurosawa, Takeshi III-275
 Kusztina, Emma III-510, III-568
 Kuwabara, Kazuhiro I-326
- Lakhal, Lotfi I-590
 Laosinchai, Parames II-134
 Lee, Chun-Jen II-285
 Lee, Gyeyoung II-203
 Lee, Hyungoo I-289
 Lee, Seongjoon I-289
 Lee, Shawn I-260
 Leimstoll, Uwe II-51
 León, Coromoto I-32
 Leray, Philippe II-176
 Leshcheva, Irina A. I-337
 Leung, Carson K.-S. II-355
 L'Huillier, Gaston II-11
 Li, Li I-424
 Li, Wei III-167
 Li, You III-217
 Li, Zhang I-528
 Lin, Mu Fei III-558
 Liu, Kokutan II-366
 Liwicki, Marcus IV-187, IV-204, IV-222
 Lokman, Gürcan I-90
 Lovrek, Ignac I-357
 Lu, Chung-Li II-285
 Luckner, Marcin IV-435
- Ludwiszewski, Bohdan II-657
 Lukose, Dickson III-346
- Maass, Wolfgang I-387
 Madokoro, Hirokazu I-446
 Maeda, Keita III-637
 Maekawa, Yasuko IV-280
 Magnani, Lorenzo II-486
 Majima, Yukie IV-280
 Makris, Dimitrios I-347
 Malerba, Donato II-559
 Mamadolimova, Aziza III-346
 Mancilla-Amaya, Leonardo II-621
 Mannweiler, Christian IV-146
 Marmann, Frank II-430
 Martínez, Luis II-124
 Marxen, Henning IV-177
 Masciari, Elio II-571
 Massey, Louis II-1
 Matsubara, Takashi III-627
 Matsuda, Noriyuki III-49
 Matsumoto, Chieko III-328
 Matsumoto, Hideyuki III-405
 Matsumoto, Kazunori III-285, IV-271
 Matsuno, Tomoaki III-106
 Matsuodani, Tohru III-336
 Matsushima, Hiroshi IV-89
 Matsuura, Kenji III-21, III-520
 Maus, Heiko II-430, IV-212
 Meixner, Gerrit IV-136
 Mejía-Gutiérrez, Ricardo II-601
 Memmel, Martin I-495, IV-126
 Methlouthi, Ines II-325
 Metz, Daniel III-56
 Miaoulis, Georgios I-347
 Mihai, Gabriel I-505
 Minaei-Bidgoli, Behrouz I-21, I-110, II-214
 Minarik, Milos I-11
 Mine, Tsunenori II-154
 Mineno, Hiroshi III-106, III-227
 Mitsuda, Takao II-366
 Miura, Hirokazu III-49
 Miura, Motoki III-96, III-539
 Miyachi, Taizo III-1, III-11
 Miyaji, Isao III-86
 Miyamoto, Takao IV-271
 Mizuno, Shinji III-548
 Mizuno, Tadanori III-106, III-207
 Mori, Hiroyuki III-405

- Mori, Yoko III-126
 Morioka, Yuichi IV-98
 Morita, Takeshi III-246
 Mukai, Naoto III-606
 Müller, Ulf III-56
 Munemori, Jun III-77, III-126, III-167
 Murai, Soichi IV-24
 Muramatsu, Kousuke III-529
 Mustapha, Nesrine Ben II-538
 Mutoh, Kouji II-447
 Myriam, Hadjouni II-41
- Nagata, Ryo II-144
 Nakagawa, Masaru III-153
 Nakahara, Takanobu III-295
 Nakahira, Katsuko T. III-499
 Nakamura, Yu IV-261
 NanakoTakata III-548
 Nasser, Youssef III-116
 Németh, Erzsébet III-385
 Nguyen, Hoai-Tuong II-176
 Nguyen, Ngoc Thanh I-210
 Niimura, Masaaki III-529
 Ninn, Kou II-366
 Nishide, Tadashi III-77
 Nishihara, Yoko II-469, III-265
 Nishino, Kazunori III-548
 Nishino, Tomoyasu III-40
- Noda, Masaru III-415
 Nomiya, Hiroki IV-1
 Nonaka, Yuki III-587
 Nunez Rattia, Rodrigo III-499
 Nyu, Takahiro III-96
- Oberreuter, Gabriel II-11
 Oehlmann, Ruediger II-440
 Ogata, Hiroaki III-520
 Ohira, Yuki IV-1
 Ohmura, Hayato IV-53
 Ohmura, Hiroaki II-21
 Ohsawa, Yukio II-469
 Okada, Yoshihiro IV-63
 Okamoto, Masayuki III-67
 Okamoto, Ryo II-165
 Okamoto, Takeshi IV-298
 Oku, Kenta IV-44
 Okubo, Yoshiaki II-457
 Olarte, Juan Gabriel II-306
 Oltean, Gabriel III-443
 Omachi, Shinichiro IV-204
- Onishi, Rie III-144
 Onozato, Taishi I-280
 Oosuka, Ryuuji III-106
 Orłowski, Aleksander II-650
 Orłowski, Cezary II-677
 Orozco, Jorge I-100
 Osogami, Masahiro I-296
 Otsuka, Shinji III-21, III-520
 Ouziri, Mourad I-548
 Ozaki, Masahiro IV-80
- Pagnotta, Stefano M. I-568
 Pan, Rong II-505
 Panjaburee, Patcharin II-134
 Parra, Carlos II-611
 Parvin, Hamid I-21, I-110, II-214
 Pellier, Damien I-548
 Pertiwi, Anggi Putri I-52
 Petrakis, Euripides G.M. I-558
 Petre, Emil IV-388
 Pfister, Thomas IV-167
 Pham, Tuan D. I-466
 Pichanachon, Akawuth I-230
 Pietranik, Marcin I-210
 Plemenos, Dimitri I-347
 Poetzsch-Heffter, Arnd IV-126
 Prickett, Paul III-433, IV-425
- Rakus-Andersson, Elisabeth IV-399
 Ramirez-Iniguez, Roberto III-453, III-462, III-472
 Ramstein, Gérard II-176
 Refanidis, Ioannis II-114
 Ren, Fuji I-456
 Resta, Marina II-372
 Ríos, Sebastián A. II-11
 Rivera, Fernando II-306
 Ro, Kou II-366
 Rombach, Dieter IV-136
 Rostanin, Oleg II-410
 Roth, Michael IV-366
 Rouhizadeh, Masoud IV-378
 Rousselot, François II-345, IV-445
 Różewski, Przemysław III-510, III-568
 Ruiz-Arenas, Santiago II-601
 Rumyantseva, Maria N. I-337
 Rybakov, Vladimir V. I-171, I-306, II-478
 Rygielski, Piotr II-591, II-687

- Saga, Ryosuke III-285, IV-271
 Saito, Muneyoshi III-356
 Sakamoto, Yuuta III-86
 Sanchez, Eider II-611
 Sanín, Cesar II-621, II-631, II-667
 Sapozhnikova, Elena I-579
 Sarlin, Peter II-186
 Sasaki, Kazuma IV-357
 Sasaki, Kenta III-67
 Sato, Hiroshi III-627
 Sato, Kazuhito I-446
 Satou, Yuuki III-86
 Sauter, Rolf II-275
 Schaaf, Marc II-51, II-275
 Schäfer, Walter III-56
 Schirru, Rafael I-495
 Schmidt, Benedikt II-402
 Schmidt, Karsten IV-126
 Schneider, Jörg IV-146
 Schneider, Klaus IV-167
 Schotten, Hans D. IV-146
 Schuldes, Stephanie IV-366
 Schwarz, Sven II-420, II-430
 Sedziwy, Adam I-180, I-190
 Segredo, Eduardo I-32
 Segura, Carlos I-32
 Seissler, Marc IV-136
 Sekanina, Lukas I-11
 Selișteanu, Dan IV-388
 Sendrescu, Dorin IV-388
 Seta, Kazuhisa III-558, IV-261, IV-288
 Shida, Haruki IV-298
 Shigeno, Aguri II-31
 Shigeyoshi, Hiroki IV-242
 Shiizuka, Hisao III-197
 Shim, Kyubark II-203
 Shimada, Satoshi IV-280
 Shimada, Yukiyasu III-423
 Shimogawa, Shinsuke III-275
 Shintani, Munehiro I-260
 Shiraishi, Soma IV-195
 Shirota, Yukari II-73
 Sikora, Katarzyna III-568
 Sikorski, Marcin II-657
 Sirola, Miki II-196
 Şişman, Zeynep I-271
 Sitarek, Tomasz IV-232
 Sitek, Tomasz II-677
 Sklavakis, Dimitrios II-114
 Slimani, Yahya I-590
 Soga, Masato III-40
 Son, Hongkwan I-289
 Söser, Peter IV-455
 Sproat, Richard IV-378
 Stanescu, Liana I-505
 Stefanoiu, Dan I-72
 Stratulat, Florin I-72
 Stratz, Alex II-275
 Stravoskoufos, Kostas I-558
 Strube, Michael IV-366
 Su, Ja-Hwung II-285
 Sugihara, Taro III-539
 Sunayama, Wataru III-265
 Suyanto I-52
 Suzuki, Motoyuki I-456
 Suzuki, Nobuo III-378
 Świątek, Paweł II-687
 Szczerbicki, Edward II-621, II-631,
 II-640, II-650, II-667
 Szpyrka, Marcin I-180, I-190
 Tagashira, Shigeaki IV-252
 Taguchi, Ryosuke III-499
 Takahashi, Masakazu III-320
 Takahiro, Masui III-106
 Takai, Keiji III-304
 Takano, Shigeru IV-63
 Takeda, Kazuhiro III-415
 Takeda, Yasuchika IV-108
 Takeshima, Syujo III-49
 Takeuchi, Shin IV-89
 Taki, Hirokazu III-40, III-49
 Takimoto, Munehiro I-260
 Takubo, Yuto III-1
 Talonen, Jaakko II-196
 Tamano, Keniti IV-242
 Tamura, Hitoshi I-280
 Tanaka, Hidekazu IV-98
 Tanaka, Katsumi II-63
 Tanaka, Kouji III-328
 Tanaka, Toshio III-21, III-520
 Tanida, Akihide I-484
 Thieme, Sandra IV-187
 Ting, Lan I-528
 Todorov, Konstantin I-538
 Tokumitsu, Masahiro IV-318
 Tomczak, Jakub M. II-591
 Topuz, Vedat I-90
 Toro, Carlos II-611
 Torsello, M.A. II-84

- Tran, Dat I-436
 Tran, Trong Hieu I-200
 Trapp, Mario IV-167
 Tschumitschew, Katharina I-82
 Tseng, Vincent S. II-285
 Tsuchiya, Seiji I-367, I-456, I-474
 Tsuda, Kazuhiko III-320, III-328,
 III-336, III-378
 Tsuji, Hiroshi IV-242, IV-261, IV-271
 Tung, Ta Son IV-44
 Uchida, Seiichi IV-195, IV-204
 Ueda, Takuya IV-338
 Ueno, Tsuyoshi IV-242
 Uetsuki, Keiji III-336
 Umano, Motohide IV-288
 Unno, Masaru III-310
 Uosaki, Katsuji I-296
 Ushijima, Taketoshi IV-24
 Utsumi, Yuya I-446
 Velásquez, Juan D. II-11
 Villarreal, Vladimir II-296
 Vo, Quoc Bao I-200
 Voiculescu, E. III-493
 Vukovic, Marin I-357
 Wang, Bo III-217
 Wang, Hui I-161
 Wang, Peng II-631
 Wanichsan, Dechawut II-134
 Watabe, Hirokazu I-367, I-474
 Watada, Junzo III-187, III-217, III-237
 Watanabe, Nayuko III-67
 Watanabe, Shosuke III-1
 Watanabe, Toyohide II-104, III-578,
 III-637, IV-14
 Wathanathamsiri, Sakon I-230
 Wehn, Norbert IV-177
 Werner-Stark, Ágnes III-385
 Wolff, Daniela II-51
 Woodham, Robert J. IV-108
 Wu, Juiyu III-237
 Wyrwiński, Jan II-657
 Xu, Guandong II-505
 Xu, Hua III-310
 Xu, Yanhao I-424
 Yaakob, Shamshul Bahar III-187
 Yada, Katsutoshi III-295, III-304
 Yamada, Keiji II-63
 Yamada, Kunihiro III-86, III-207,
 III-227
 Yamagiwa, Shinichi III-21
 Yamaguchi, Takahira III-246
 Yamanishi, Teruya I-296
 Yamano, Takayuki I-220
 Yamazaki, Atsuko K. III-31
 Yan, Wei II-345, IV-445
 Yano, Yoneo III-21, III-520
 Yasunaga, Shotaro I-326
 Yim, Jaeggeol II-203
 Yinwen, Zhang I-528
 Yoshida, Akira IV-204
 Yoshida, Kaori III-177
 Yoshida, Kouji III-86, III-144, III-207
 Yoshihiro, Takuya III-153
 Yoshimura, Eriko I-367, I-474
 Yoshino, Takashi I-220, II-31
 Yuizono, Takaya III-167
 Yusa, Naoki III-227
 Zanni-Merk, Cecilia II-345, IV-445
 Zatwarnicka, Anna I-141
 Zatwarnicki, Krzysztof I-42, I-141
 Zghal, Hajar Baazaoui II-538
 Zhang, Haoxi II-667
 Zhang, Xicen III-578
 Zong, Yu II-505
 Zühlke, Detlef IV-136