Advances in Intelligent Systems and Computing 620

Sigeru Omatu
Sara Rodríguez
Gabriel Villarrubia
Pedro Faria
Paweł Sitek
Javier Prieto *Editors*

Distributed Computing and Artificial Intelligence, 14th International Conference

Springer

Advances in Intelligent Systems and Computing

Volume 620

Series editor

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

About this Series

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

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

Advisory Board

Chairman

Nikhil R. Pal, Indian Statistical Institute, Kolkata, India
e-mail: nikhil@isical.ac.in

Members

Rafael Bello Perez, Universidad Central "Marta Abreu" de Las Villas, Santa Clara, Cuba
e-mail: rbellop@uclv.edu.cu

Emilio S. Corchado, University of Salamanca, Salamanca, Spain
e-mail: escorchado@usal.es

Hani Hagras, University of Essex, Colchester, UK
e-mail: hani@essex.ac.uk

László T. Kóczy, Széchenyi István University, Győr, Hungary
e-mail: koczy@sze.hu

Vladik Kreinovich, University of Texas at El Paso, El Paso, USA
e-mail: vladik@utep.edu

Chin-Teng Lin, National Chiao Tung University, Hsinchu, Taiwan
e-mail: ctlin@mail.nctu.edu.tw

Jie Lu, University of Technology, Sydney, Australia
e-mail: Jie.Lu@uts.edu.au

Patricia Melin, Tijuana Institute of Technology, Tijuana, Mexico
e-mail: epmelin@hafsamx.org

Nadia Nedjah, State University of Rio de Janeiro, Rio de Janeiro, Brazil
e-mail: nadia@eng.uerj.br

Ngoc Thanh Nguyen, Wroclaw University of Technology, Wroclaw, Poland
e-mail: Ngoc-Thanh.Nguyen@pwr.edu.pl

Jun Wang, The Chinese University of Hong Kong, Shatin, Hong Kong
e-mail: jwang@mae.cuhk.edu.hk

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

Sigeru Omatu · Sara Rodríguez
Gabriel Villarrubia · Pedro Faria
Paweł Sitek · Javier Prieto
Editors

Distributed Computing and Artificial Intelligence, 14th International Conference

Springer

Editors
Sigeru Omatu
Faculty of Engineering, Department of Electronics, Information and Communication Engineering
Osaka Institute of Technology
Osaka, Osaka
Japan

Sara Rodríguez
Faculty of Science, Department of Computer Science and Automation Control
University of Salamanca
Salamanca
Spain

Gabriel Villarrubia
Faculty of Science, Department of Computer Science and Automation Control
University of Salamanca
Salamanca
Spain

Pedro Faria
GECAD – Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development
Instituto Superior de Engenharia do Porto
Porto
Portugal

Paweł Sitek
Faculty of Electrical Engineering and Computer Science, Department of Control and Management Systems, Department of Information Systems
Kielce University of Technology
Kielce
Poland

Javier Prieto
Faculty of Science, Department of Computer Science and Automation Control
University of Salamanca
Salamanca
Spain

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-62409-9 ISBN 978-3-319-62410-5 (eBook)
DOI 10.1007/978-3-319-62410-5

Library of Congress Control Number: 2017944919

© Springer International Publishing AG 2018
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Nowadays, most computing systems from personal laptops/computers to cluster/grid/cloud computing systems are available for parallel and distributed computing. Distributed computing performs an increasingly important role in modern signal/data processing, information fusion and electronics engineering (e.g. electronic commerce, mobile communications and wireless devices). Particularly, applying artificial intelligence in distributed environments is becoming an element of high added value and economic potential. Research on Intelligent Distributed Systems has matured during the last decade and many effective applications are now deployed. The artificial intelligence is changing our society. Its application in distributed environments, such as the Internet, electronic commerce, mobile communications, wireless devices, distributed computing, and so on is increasing and is becoming an element of high added value and economic potential, both industrial and research. These technologies are changing constantly as a result of the large research and technical effort being undertaken in both universities and businesses.

The 14th International Symposium on Distributed Computing and Artificial Intelligence 2017 (DCAI 2017) is a forum to present applications of innovative techniques for solving complex problems in these areas. The exchange of ideas between scientists and technicians from both academic and business areas is essential to facilitate the development of systems that meet the demands of today's society. The technology transfer in this field is still a challenge and for that reason this type of contributions will be specially considered in this symposium. This conference is the forum in which to present application of innovative techniques to complex problems. This year's technical program will present both high quality and diversity, with contributions in well-established and evolving areas of research. Specifically, 58 papers were submitted from over 21 different countries (Angola, Austria, Belgium, Brazil, China, Colombia, Denmark, Ecuador, France, India, Ireland, Italy, Japan, Malaysia, Mexico, Poland, Portugal, Spain, Switzerland, Taiwan, Tunisia, USA), representing a truly "wide area network" of research activity. The DCAI'17 technical program has selected 41 papers and, as in past editions, it will be special issues in JCR-ranked journals such as Neurocomputing, and International Journal of Knowledge and Information Systems. These special

issues will cover extended versions of the most highly regarded works. Moreover, DCAI'17 Special Sessions have been a very useful tool in order to complement the regular program with new or emerging topics of particular interest to the participating community. Special Sessions that emphasize on multi-disciplinary and transversal aspects, such as *AI-driven methods for Multimodal Networks and Processes Modeling* and *Intelligent and Secure Management towards Smart Buildings and Smart Grids* have been especially encouraged and welcome.

This symposium is organized by the Polytechnic of Porto, the Osaka Institute of Technology and the University of Salamanca. The present edition was held in Porto, Portugal, from 21–23rd June, 2017.

We thank the sponsors (IBM, Indra, IEEE Systems Man and Cybernetics Society Spain) and the funding supporting of the Junta de Castilla y León (Spain) with the project *"Moviurban: Máquina Social para la Gestión sostenible de Ciudades Inteligentes: Movilidad Urbana, Datos abiertos, Sensores Móviles"* (ID. SA070U16 – Project co-financed with FEDER funds), and finally, the Local Organization members and the Program Committee members for their hard work, which was essential for the success of DCAI'17.

Osaka, Japan — Sigeru Omatu
Salamanca, Spain — Sara Rodríguez
Salamanca, Spain — Gabriel Villarrubia
Kielce, Poland — Pedro Faria
Porto, Portugal — Paweł Sitek
Salamanca, Spain — Javier Prieto
June 2017

Organization

Honorary Chairman

- Masataka Inoue, President of Osaka Institute of Technology, Japan

Program Committee Chairs

- Sigeru Omatu, Osaka Institute of Technology, Japan
- Sara Rodríguez, University of Salamanca, Spain
- Fernando De la Prieta, University of Salamanca, Spain

Local Committee Chair

- Pedro Faria, Instituto Superior de Engenharia do Porto, Portugal

Organizing Committee

- Filipe Sousa, Polytechnic of Porto, Portugal
- Gabriel Santos, Polytechnic of Porto, Portugal
- João Soares, Polytechnic of Porto, Portugal
- João Spínola, Polytechnic of Porto, Portugal
- Luís Conceição, Polytechnic of Porto, Portugal
- Nuno Borges, Polytechnic of Porto, Portugal
- Sérgio Ramos, Polytechnic of Porto, Portugal
- Christian Retire, Université de Montpellier and LIRMM-CNRS, France

Scientific Committee

- Silvana Aciar, Instituto de Informática, Universidad Nacional de San Juan, Argentina
- Ana Almeida, GECAD-ISEP-IPP, Portugal
- Gustavo Almeida, Instituto Federal do Espírito Santo, Brazil
- Giner Alor Hernandez, Instituto Tecnologico de Orizaba, Mexico

- Cesar Analide, University of Minho, Portugal
- Luis Antunes, GUESS/LabMAg/Universidade de Lisboa, Portugal
- Fidel Aznar, Universidad de Alicante, Spain
- Javier Bajo, Universidad Politécnica de Madrid, Spain
- Zbigniew Banaszak, Faculty of Management, Department of Business Informatics, Warsaw University of Technology, Poland
- Olfa Belkahla Driss, Institut Supérieur de Gestion de Tunis (labo SOIE), Tūnis
- Óscar Belmonte, Jaume I University, Spain
- Carmen Benavides
- Enrique J. Bernabeu, Universitat Politècnica de Valencia, Spain
- Xiomara Patricia Blanco Valencia, BISITE, Spain
- Amel Borgi, ISI/LIPAH, Université de Tunis El Manar, Tūnis
- Pierre Borne, Ecole Centrale de Lille, France
- Lourdes Borrajo, University of Vigo, Spain
- Adel Boukhadra, National high School of Computer Science, Oued-Smar, Algeria
- Edgardo Bucciarelli, University of Chieti-Pescara, Italy
- Francisco Javier Calle, Departamento de Informática, Universidad Carlos III de Madrid, Spain
- Rui Camacho
- Juana Canul Reich, Universidad Juarez Autonoma de Tabasco, Mexico
- Davide Carneiro
- Carlos Carrascosa, GTI-IA DSIC Universidad Politecnica de Valencia, Spain
- Camelia Chira, Technical University of Cluj-Napoca, Romania
- Rafael Corchuelo, University of Seville, Spain
- Paulo Cortez, University of Minho, Portugal
- Ângelo Costa, Universidade do Minho, Portugal
- Stefania Costantini, Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica, University of L'Aquila, Italy
- Jamal Dargham, Universiti Malaysia Sabah, Malaysia
- Giovanni De Gasperis, Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica, Italy
- Carlos Alejandro De Luna-Ortega, Universidad Politecnica de Aguascalientes, Mexico
- Raffaele Dell'Aversana, Research Centre for Evaluation and Socio-Economic Development, Italy
- Fernando Diaz, University of Valladolid, Spain
- Worawan Diaz Carballo, Thammasat University, Thailand
- Youcef Djenouri, LRIA_USTHB, Algérie
- Ramon Fabregat, Universitat de Girona, Spain
- Johannes Fähndrich, Technische Universität Berlin/DAI Labor, Germany
- Ana Faria, ISEP, Portugal
- Florentino Fdez-Riverola, University of Vigo, Spain
- Alberto Fernandez CETINIA, University Rey Juan Carlos, Spain
- João Ferreira, ISCTE, Portugal

- Peter Forbrig, University of Rostock, Germany
- Felix Freitag, UPC, Spain
- Toru Fujinaka, Hiroshima University, Japan
- Donatella Furia, University of Chieti-Pescara, Italy
- Svitlana Galeshchuk, Nova Southeastern University (EE.UU)
- Francisco Garcia-Sanchez, University of Murcia, Spain
- Marisol García-Valls, University Carlos III of Madrid, Spain
- Irina Georgescu, Academy of Economic Studies, Romania
- Ana Belén Gil González, University of Salamanca, Spain
- Federico Gobbo
- Juan Gomez Romero, Universidad de Granada, Spain
- Evelio Gonzalez, Universidad de La Laguna, Spain
- Carina Gonzalez
- Angélica González Arrieta, Universidad de Salamanca, Spain
- David Griol, Universidad Carlos III de Madrid, Spain
- Samer Hassan, Universidad Complutense de Madrid, Spain
- Daniel Hernández de La Iglesia, Universidad de Salamanca, Spain
- Felipe Hernández Perlines, Universidad de Castilla-La Mancha, Spain
- Elisa Huzita, State University of Maringa, Brazil
- Gustavo Isaza, University of Caldas, Colombia
- Patricia Jiménez, Universidad de Huelva, Spain
- Bo Noerregaard Joergensen, University of Southern Denmark, Denmark
- Vicente Julian, GTI-IA DSIC UPV, Spain
- Amin Khan, Universitat Politecnica de Catalunya, Spain
- Naoufel Khayati, COSMOS Laboratory—ENSI, Tūnis
- Egons Lavendelis, Riga Technical University, Letonia
- Rosalia Laza, Universidad de Vigo, Spain
- Tiancheng Li, Northwestern Polytechnical University, China
- Johan Lilius, Åbo Akademi University, Finland
- Dolores Mª Llidó Escrivá, University Jaume I, Spain
- Faraón Llorens-Largo, Universidad de Alicante, Spain
- Alberto López Barriuso, University of Salamanca, Spain
- Ivan Lopez-Arevalo Cinvestav, Tamaulipas, Mexico
- Jose J. Lopez-Espin, Universidad Miguel Hernández de Elche, Spain
- Roussanka Loukanova, Stockholm University, Sweden
- Álvaro Lozano Murciego, USAL, Spain
- Ramdane Maamri, LIRE laboratory UC Constantine2-Abdelhamid Mehri, Algérie
- Moamin Mahmoud, Universiti Tenaga Nasional, Malaysia
- Benedita Malheiro, Instituto Superior de Engenharia do Porto, Portugal
- Eleni Mangina, UCD, Ireland
- Fabio Marques, University of Aveiro, Portugal
- Goreti Marreiros, University of Minho, Portugal
- Angel Martin Del Rey, Department of Applied Mathematics, Universidad de Salamanca, Spain

- Ester Martinez-Martin, Universitat Jaume I, Spain
- Philippe Mathieu, University of Lille 1, France
- Sérgio Matos, IEETA, Universidade de Aveiro, Portugal
- Kenji Matsui, Osaka Institute of Technology, Japan
- Jacopo Mauro, University of Oslo, Norway
- Mohamed Arezki Mellal, M'Hamed Bougara University, Algérie
- Fethi Mguis, Faculty of Sciences of Gabes, Tūnis
- Heman Mohabeer, Charles Telfair Institute, Maurice
- Mohd Saberi Mohamad, Universiti Teknologi Malaysia, Malaysia
- Jose M. Molina, Universidad Carlos III de Madrid, Spain
- Miguel Molina-Solana, Department of Computing, Imperial College London, UK
- Richard Moot, CNRS (LaBRI) & Bordeaux University, France
- A. Jorge Morais, Universidade Aberta, Portugal
- Antonio Moreno, URV, Spain
- Paulo Mourao, University of Minho, Portugal
- Muhammad Marwan Muhammad Fuad, Aarhus University, Denmark
- Eduardo Munera, Institute of Control Systems and Industrial Computing, Polytechnic University of Valencia, Spain
- Susana Muñoz Hernández, Technical University of Madrid, Spain
- María Navarro, BISITE, Spain
- Jose Neves, Universidade do Minho, Portugal
- Julio Cesar Nievola, Pontifícia Universidade Católica do Paraná—PUCPR Programa de Pós Graduação em Informática Aplicada, Brazil
- Nadia Nouali-Taboudjemat, CERIST, Algérie
- Paulo Novais, University of Minho, Portugal
- C. Alberto Ochoa-Zezatti, Universidad Autónoma de Ciudad Juárez, Mexico
- Paulo Moura Oliveira, UTAD University, Portugal
- José Luis Oliveira, University of Aveiro, Portugal
- Jordi Palacin, Universitat de Lleida, Spain
- Miguel Angel Patricio, Universidad Carlos III de Madrid, Spain
- Juan Pavón, Universidad Complutense de Madrid, Spain
- Reyes Pavón, University of Vigo, Spain
- Pawel Pawlewski, Poznan University of Technology, Poland
- Diego Hernán Peluffo-Ordoñez, Universidad Técnica del Norte, Ecuador
- Stefan-Gheorghe Pentiuc, University Stefan cel Mare Suceava, Romania
- Antonio Pereira, Escola Superior de Tecnologia e Gestão do IPLeiria, Portugal
- Tiago Pinto, Polytechnic of Porto, Portugal
- Julio Ponce, Universidad Autónoma de Aguascalientes, Mexico
- Juan-Luis Posadas-Yagüe, Universitat Politècnica de València, Spain
- Jose-Luis Poza-Luján, Universitat Politècnica de València, Spain
- Isabel Praça, GECAD/ISEP, Portugal
- Radu-Emil Precup, Politehnica University of Timisoara, Romania
- Mar Pujol, Universidad de Alicante, Spain

- Francisco A. Pujol, Specialized Processor Architectures Lab, DTIC, EPS, University of Alicante, Spain
- Mariano Raboso Mateos, Facultad de Informática, Universidad Pontificia de Salamanca, Spain
- Miguel Rebollo, Universidad Politécnica de Valencia, Spain
- Manuel Resinas, University of Seville, Spain
- Jaime Andres Rincon Arango, UPV, Spain
- Ramon Rizo, Universidad de Alicante, Spain
- Sergi Robles, Universitat Autònoma de Barcelona, Spain
- Cristian Aaron Rodriguez Enriquez, Instituto Tecnológico de Orizaba, Mexico
- Luiz Romao, Univille, Brazil
- Gustavo Santos-Garcia, Universidad de Salamanca, Spain
- Ichiro Satoh, National Institute of Informatics, Japan
- Ken Satoh, National Institute of Informatics and Sokendai, Japan
- Yann Secq, LIFL Lille1, France
- Ali Selamat, Universiti Teknologi Malaysia, Malaysia
- Carolina Senabre, Universidad Miguel Hernández, Spain
- Emilio Serrano, Universidad Politécnica de Madrid, Spain
- Fábio Silva, Universidade do Minho, Portugal
- Nuno Silva, DEI & GECAD, ISEP IPP, Portugal
- Paweł Sitek, Kielce University of Technology, Poland
- Pedro Sousa, University of Minho, Portugal
- Y.C. Tang, University of Tenaga Nasional, Malaysia
- Leandro Tortosa, University of Alicante, Spain
- Volodymyr Turchenko Research, Institute for Intelligent Computing Systems, Ternopil National Economic University, Ucrania
- Miki Ueno, Toyohashi University of Technology, Japan
- Zita Vale GECAD, ISEP/IPP, Portugal
- Rafael Valencia-Garcia, Departamento de Informática y Sistemas, Universidad de Murcia, Spain
- Miguel A. Vega-Rodríguez, University of Extremadura, Spain
- Maria João Viamonte, Instituto Superior de Engenharia do Porto, Portugal
- Paulo Vieira, Instituto Politécnico da Guarda, Portugal
- Jørgen Villadsen, Technical University of Denmark, Denmark
- José Ramón Villar, University of Oviedo, Spain
- Friederike Wall, Alpen-Adria-Universitaet Klagenfurt, Austria
- Zhu Wang, XINGTANG Telecommunications Technology Co., Ltd.
- Li Weigang, University of Brasilia, Brazil
- Morten Gill Wollsen, University of Southern Denmark, Denmark
- Bozena Wozna-Szczesniak, Institute of Mathematics and Computer Science, Jan Dlugosz University in Czestochowa, Poland
- Michal Wozniak, Wroclaw University of Technology, Poland
- Takuya Yoshihiro, Faculty of Systems Engineering, Wakayama University, Japan
- Michifumi Yoshioka, Osaka Prefecture University, Japan

- Agnieszka Zbrzezny, Institute of Mathematics and Computer Science, Jan Dlugosz University in Czestochowa, Poland
- Andrzej Zbrzezny, Institute of Mathematics and Computer Science, Jan Dlugosz University in Czestochowa, Poland
- André Zúquete, IEETA, University of Aveiro, Portugal

AI–driven methods for Multimodal Networks and Processes Modeling Special Session Committee

Chairs

- Paweł Sitek, Kielce University of Technology, Poland
- Grzegorz Bocewicz, Koszalin University of Technology, Poland
- Izabela E. Nielsen, Aalborg University, Denmark

Co-Chairs

- Peter Nielsen, Aalborg University, Denmark
- Zbigniew Banaszak, Warsaw University of Technology, Poland
- Paweł Pawlewski, Poznan University of Technology, Poland
- Mukund Nilakantan, Aalborg University, Denmark
- Robert Wójcik, Wrocław University of Technology, Poland
- Marcin Relich, University of Zielona Gora, Poland
- Arkadiusz Gola, Lublin University of Technology, Poland

Intelligent and Secure Management Towards Smart Buildings and Smart Grids Special Session Committee

Organizing Committee

- Isabel Praça, Polytechnic of Porto, School of Engineering (ISEP), Portugal
- Adrien Becue, Airbus Defence and Space Cybersecurity, France
- Zita Vale, Polytechnic of Porto—School of Engineering (ISEP), Portugal
- Meritxell Vinyals, CEA, France

Program Committee

- Amin Shokri, USAL, Spain
- André Richter, OVGU, Germany
- Dante Tapia, Nebusens, Spain
- Guillaume Habault, IMT, France
- Hassan Sleiman, CEA, France
- Helia Pouyllau, TRT, France
- Hugo Morais, EDF, France

- Isabel Praça, IPP, Portugal
- Jerome Rocheteau, ICAM, France
- Lamya Belhaj, ICAM, France
- Maxime Lefrançois, Armines, France
- Maxime Velay, CEA, France
- Meritxell Vinyals, CEA, France
- Noël Crespi, IMT, France
- Óscar García, Nebusens, Spain
- Pedro Faria, IPP, Portugal
- Philippe Calvez, Engie, France
- Samuel Abreu, IFSC, Brasil
- Sandra G. Rodriguez, CEA, France
- Serge Dedeystere, Institute Mines Telecom
- Sidi-Mohammed Senouci, UB, France
- Tiago Pinto, IPP, Portugal
- Tiago Soares, DTU, Denmark
- Zita Vale, IPP, Portugal

Contents

Part I
AI–driven methods for Multimodal Networks and Processes Modeling

Optimization of urban freight distribution with different time constraints - a hybrid approach

Paweł Sitek, Jarosław Wikarek, Tadeusz Stefański

Department of Control and Management Systems, Kielce University of Technology, Poland
e-mail: {sitek, j.wikarek, t.stefanski}@tu.kielce.pl

Abstract. The efficient and timely distribution of freight is critical for supporting the demands of modern urban areas. Without optimal freight distribution, urban areas could not survive and develop. The paper presents the concepts of hybrid approach to optimization of urban freight distribution. This approach proposed combines the strengths of mathematical programming (MP) and constraint logic programming (CLP), which leads to a significant reduction in the search time necessary to find the optimal solution and allows solving larger problems. It also presents the formal model for optimization of urban freight distribution with different types of time constraints. The application of the hybrid approach to the optimization of urban freight distribution is the primary contribution of this paper. The proposed model was implemented using both the hybrid approach and pure mathematical programming for comparison. Several experiments were performed for both computational implementations in order to evaluate both approaches.

Keywords: Urban Freight Distribution, Hybrid methods, Constraint Logic Programming, Mathematical Programming, Optimization, Presolving.

1 Introduction

Urban Freight Distribution (UFD) is the system and set of processes by which consumer goods are collected, packed, transported, and distributed within urban environments [1]. In general, the urban freight system can include airports, seaports, manufacturing facilities, and distribution centers/warehouses that are connected by a network railroads, roadways, highways and pipelines, that enable goods to get to their destinations. The principal challenge of UFD is the efficient and timely distribution of freight for supporting the demands of modern urban areas and functions. UFD is essential to supporting and developing international and domestic trade as well as the daily needs of local businesses and consumers. In addition, it provides large number of jobs and other economic benefits [2]. The problem of urban freight distribution is characterized by a large number of different constraints. Constraints concern the resources, capacity, transportation, time and so on. Particularly important in this context are time constraints. They relate to on-time delivery, transport time, working time of drivers and so on [3]. The main contribution of this research is the hybrid approach to the modeling and optimization of urban freight distribution problem. The formal op-

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_1

timization model of the UFD problem was presented along with computational experiments for this model using the hybrid approach [3,4] and mathematical programming [5]. The application of the hybrid approach to the optimization of urban freight distribution is the primary contribution of this paper. The approach proposed offers: (a) finding optimal solution in a substantially shorter time than that needed by mathematical programming, (b) solving larger size models and (c) modeling other e.g., nonlinear and logic constraints, in addition to linear constraints.

2 Problem description

There is a UFD system that consists of depots/distribution center d ($d \in D$, D–the set of all depots). These depots supply delivery points (retailers, shops, etc.) o ($o \in O$, O–the set of all delivery points) by sending them shipments s ($s \in S$, S–the set of all shipments). A shipment may be a pallet / a bin /a package containing a variety of products for the delivery point. The depot can send some shipments (due to e.g. the maximum size of pallet / bin / packages) to the delivery point. The delivery point can get shipments from several depots. These shipments are delivered by courier / driver k ($k \in K$, K–the set of all couriers / drivers). Couriers are assigned to a specific depot / distribution center (due to e.g. the specific nature of transported goods that must match the specifics of the vehicle, e.g. refrigerated truck, tank truck etc.). The courier can deliver the shipments to several delivery points in one run of the route.

Shipments have specific dimensions / sizes. The courier can only take the number of shipments such that it does not exceed the capacity of his vehicle. There are known travel times between depots and delivery points and the time it takes to unload shipment at the destination (delivery point). It is assumed that there is no possibility of reloading shipments. Specify how to realize delivery (i.e. which couriers and in which order) from the depots to the delivery points to minimize the total travel time. Parameters and decision variables for mathematical model of UFD system are presented in table 1.

There may be additional time constraints:

- working time of a courier / supplier is not greater than time T.
- some delivery to the selected delivery points must be within a certain time window.

The objective function (1) is the minimization of all traveling times. Arrival and departure from the point i are ensured by constraint (2). If no items are to be carried on the route, a courier does not travel that route (3). If a courier k does not travel along a route, no shipments are to be carried on that route (4). At no route segment, courier k carries more shipments s than the allowable courier's vehicle capacity (5). Shipping must be downloaded from the depot d (6). Constraint (7) ensures delivery of shipments to the delivery points. Each courier k picked up the shipments s from a depot d (8). Constraint (9) defines a set of shipments s for the courier k. The selected shipment is delivered by only one courier (10). Constraint (11) determines the time of arrival of the courier k to the delivery point o. Other constraints determine the binarity of decision variables.

Table 1. Parameters and decision variables for mathematical model

Symbol	Description
	Parameters
Vs_s	Shipment volume (volumetric weight) s ($s \in S$)
Vk_k	Courier's vehicle volume k ($k \in K$)
TS_o	Unloading time at delivery point o ($o \in O$)
$SDO_{s,d,o}$	If the shipment s ($s \in S$) is delivered from depot d to the delivery point o then $SDO_{s,d,o}=1$ otherwise $SDO_{s,d,o}=0$
$DK_{d,k}$	If the courier k operates from the depot d than $DK_{d,k}=1$ otherwise $DK_{d,k}=0$
$TT_{i,j}$	Travelling time from the point i ($i \in D \cup O$) to the point j ($j \in D \cup O$)
WX	A large number quantity
	Decision variables
$X_{k,i,j}$	If courier k travels from point i to point j than $X_{k,i,j}=1$, otherwise $X_{k,i,j}=0$, ($k \in K$, $i \in O \cup D$, $j \in O \cup D$)
$Y_{k,i,j,s}$	If courier k travels from point i to point j carrying shipment s than $Y_{k,i,j,s}=1$, otherwise $Y_{k,i,j,s}=0$, ($k \in K$, $i \in O \cup D$, $j \in O$, $s \in S$)
$FX_{k,s}$	If shipment s is delivered by courier k then $FX_{k,s}=1$, otherwise $FX_{k,s}=0$ ($k \in K$, $s \in S$)
$TX_{k,i}$	A moment of time in which courier k arrives to the delivery point/depot i ($k \in K$, $i \in O \cup D$)

$$\sum_{k\in K}\sum_{i\in O\cup D}\sum_{j\in O\cup D} X_{k,i,j}\cdot TT_{i,j} \tag{1}$$

$$\sum_{j\in O\cup D} X_{k,i,j} = \sum_{j\in O\cup D} X_{k,j,i} \;\forall\; k\in K, i\in O\cup D : DK_{d,k}=1 \tag{2}$$

$$X_{k,i,j} \le \sum_{s\in S} Y_{k,i,j,s} \;\forall\; k\in K, i\in O\cup D, j\in O \tag{3}$$

$$Y_{k,i,j,s} \le X_{k,i,j} \;\forall\; k\in K, i\in O\cup D, j\in O\cup D, s\in S \tag{4}$$

$$\sum_{s\in S} Vs_s\cdot Y_{k,i,j,s} \le Vk_k\cdot X_{k,i,j} \;\forall\; k\in K, i\in O\cup D, j\in O\cup D \tag{5}$$

$$\sum_{k\in K}\sum_{j\in D} Y_{k,d,j,s} = 1 \forall\; s\in S, d\in D, o\in O : SDO_{s,d,o}=1 \tag{6}$$

$$\sum_{k\in K}\sum_{j\in O\cup D} Y_{k,d,j,s} - \sum_{k\in O}\sum_{j\in O} Y_{k,o,j,s} = SDO_{s,d,o} \forall\; s\in S, d\in D, o\in O \tag{7}$$

$$\sum_{k\in K}\sum_{i\in O} Y_{k,d,i,s,g} = SDO_{s,d,o} \;\forall\; s\in S, d\in D, o\in O \tag{8}$$

$$\sum_{i\in O\cup D}\sum_{j\in O\cup D} Y_{k,d,i,s} \le WX\cdot FX_{s,k} \;\forall\; s\in S, k\in K \tag{9}$$

$$\sum_{k\in K} FX_{s,k} = 1 \;\forall\; s\in S \tag{10}$$

$$TX_{k,i} - WX\cdot(1-X_{k,i,j}) + TT_{i,j}\cdot X_{k,i,j} \le TX_{k,j} \forall\; k\in K, i\in O\cup D, j\in O \tag{11}$$

$$Y_{k,i,j,s} = \{0,1\} \ \forall \ k \in k, i \in O \cup D, j \in O \cup D, s \in S \tag{12}$$

$$X_{k,i,j} = \{0,1\} \ \forall \ k \in k, i \in O \cup D, j \in O \cup D \tag{13}$$

$$FX_{s,k} = \{0,1\} \ \forall \ s \in S, k \in K \tag{14}$$

3 Methodology – a hybrid approach

Numerous studies and our previous experience show that the constraint-based environment [6,7] offers a very good environment for representing the knowledge, information and methods needed for modeling and solving complex problems such as UFD. The constraint logic programming (CLP) is a particularly interesting option in this context. The CLP is a form of constraint programming (CP) in which logic programming is extended to include concepts from constraint satisfaction [6]. A constraint logic program contains constraints in the body of clauses (predicates). Effective search for the solution in the CLP depends considerably on the effective constraint propagation, which makes it a key method of the constraint-based approach. Constraint propagation embeds any reasoning that consists in explicitly forbidding values or combinations of values for some variables of a problem because a given subset of its constraints cannot be satisfied otherwise [7].

Based on [6-8] and our previous work [4,9,10], we observed some advantages and disadvantages of MP and CLP environments. An integrated approach of CLP and MP (mathematical programming) can help to solve optimization problems that are intractable with either of the two methods alone [11,12].

Both the MP and the finite domain CLP involve variables and constraints. However, the types of the variables and constraints that are used, and the ways the constraints are solved are different in the two approaches [5,12,13]. Both in MP and in CLP, there is a group of constraints that can be solved with ease and a group of constraints that are difficult to solve. The easily solved constraints in the MP methods are linear equations and inequalities over rational numbers. Integrity constraints are difficult to solve using mathematical programming methods and often the real problems of the MIP make them NP-hard. In the CP/CLP, domain constraints with integers and equations between variables are easy to solve. The system of such constraints can be solved over integer variables in polynomial time. The inequalities between variables, general linear constraints, and symbolic constraints are difficult to solve in the CP/CLP (NP-hard). This type of constraints reduces the strength of constraint propagation. The MP approach focuses mainly on the methods of solving and, to a lesser degree, on the structure of the problem. The data, however, is completely outside the model. The same model without any changes can be solved for multiple instances of data. In the CLP approach, due to its declarative nature, the methods are already in place. The data and structure of the problem are used for its modeling.

These observations and the knowledge of the properties of CLP and MP systems enforce the integration. Our approach differs from the known integration of CLP/MP

[12-14]. This approach, called hybridization, consists of the combination of both environments and the transformation of the model.

The transformation is seen as a presolving method. In general, it involves elimination from the model solutions space of those points which are unacceptable. It is determined on the basis of data instances and model constraints. For example, in UFD, based on analysis of instances of data stored in the form of facts, you can specify that shipments cannot be prepared by the given depots, which delivery points placed orders, etc. The general concept of the hybrid approach is shown in Figure 1.

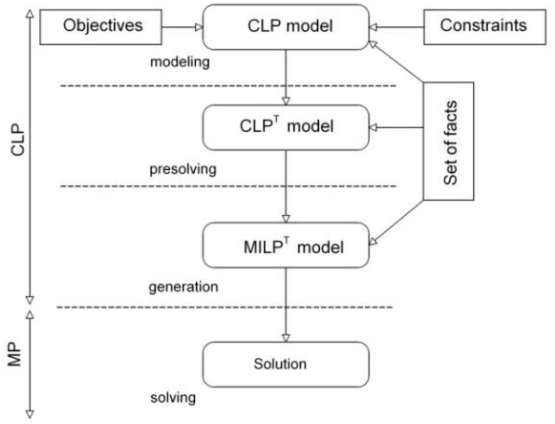


Fig. 1. The concept of the hybrid approach.

Figure 1 presents the general concept of the hybrid approach. This approach comprises several phases: modeling, presolving, generation phase and solving. It has two inputs and uses the set of facts. Inputs are the set of objectives and the set of constraints to the model of a given problem. Based on them, the primary model of the problem is generated as a CLP model, which is then presolved. The built-in CLP method (constraint propagation [7]) and the method of problem transformation designed by the authors [4,9] are used for this purpose. Presolving procedure results on the transformed model CLPT. This model is the basis for the automatic generation of the MILP (Mixed Integer Linear Programming) model, which is solved in MP (with the use of an external solver or as a library of CLP).

4 Computational experiments

All the experiments relate to the UFD with four depots (*d=1..4*), eight delivery points (*o=1..8*), eight couriers (*k=1..8*), and three sets of orders (*E(10), E(18), E(28)*). Computational experiments were carried out for the model (Section 2) implemented using mathematical programming and the hybrid approach. Experiments were made for different sets of orders and the introduction of various combination of additional time constraints *C1* (delivery by courier cannot exceed delivery time T=190 (15)) and *C2* (time window are taken into account (16)). The results are shown in Table 3 and Fig-

ure 2a, 2b. Analysis of the results clearly shows the superiority of the hybrid approach for use with UFD problems. The parameters for additional time constraints are presented in Table 2.

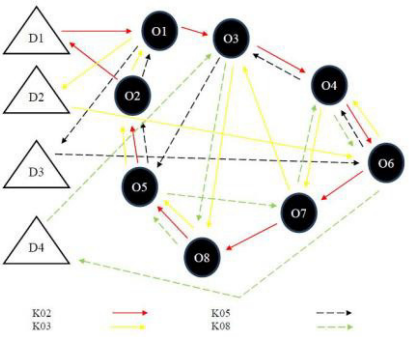


Fig. 2a. Optimal delivery network for *E9*

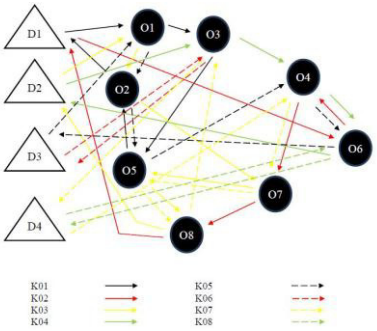


Fig. 2b. Optimal delivery network for *E12*

Table 2. Parameters for time constraints *C1* and *C2*

Symbol	Description
T	The time at which delivery must be carried out
$t1_{d,o}$	The beginning of the time window for delivery from the depot *d* to the delivery point *o*
$t2_{d,o}$	The end of the time window for delivery from the depot *d* to the delivery point *o*

$$TX_{k,i} \le T \forall\ k \in K, i \in D \tag{15}$$

$$Time\_delivery(d, o, t1_{d,o}, t2_{d,o}) \tag{16}$$

Table 3. Results of computational examples

E	Number of orders	Additional Constrains	MP				Hybrid approach			
			Fc	T	V_{int}	C	Fc	T	V	C
E1	10	---	195	341	12712	31461	195	21	1891	554
E2	10	*C1*	297	548	12712	31462	297	41	1891	555
E3	10	*C2*	195	359	12712	31469	195	22	1891	562
E4	10	*C1+C2*	302	378	12712	31470	302	48	1891	563
E5	18	---	340	541	22056	50168	340	30	2952	929
E6	18	*C1*	410	849	22056	50169	410	53	2952	930
E7	18	*C2*	349	573	22056	50176	349	33	2952	937
E8	18	*C1+C2*	425	604	22056	50177	425	36	2952	938
E9	28	---	508	755	33520	73430	508	42	4272	1416
E10	28	*C1*	659*	900**	33520	73431	659	63	4272	1417
E11	28	*C2*	509	734	33520	73438	509	48	4272	1424
E12	28	*C1+C2*	683*	900**	33520	73439	683	73	4272	1425

T	Time of finding solution (in seconds)	Fc	Objective function
V_{int}	The number of integer decision variables	C	The number of constrains
*	Feasible solution (not found optimality)	E	Example
**	Interrupt the process of finding a solution after a given time 1500 s		

Compared with the MP method, the hybrid approach substantially reduced the size of the problem solved. The number of variables was reduced up to 8 times and constraints up to 56 times. And most importantly solution time was reduced up to 18 times. Moreover, for some examples you did not get the optimal solution using the MP in an acceptable time.

A detailed analysis of the results allows determining optimal delivery times, the effect of time constraints C1(15) and C2(16) on the optimal solution, and optimal delivery networks (routes, number of couriers, etc.) (Figure 2a and Figure 2b).

5 Conclusion

This paper provides a robust and effective hybrid approach to modeling and optimization of the UFD problem. The hybrid approach incorporates two environments (i) mathematical programming and (ii) constraint logic programming. The use of the hybrid approach allows the optimization of larger size problems within a shorter time compared to mathematical programming (Table 3). It also makes it easy to model all types of constraints and store data in the form of a set of facts. The data storage in the form of a set of facts allows easy integration of the proposed hybrid approach with database systems, data warehouses, and even flat files, e.g. XML, which provides high flexibility to the hybrid approach.

In the versions to follow, implementation is planned of other additional constraints (e.g. two couriers cannot be at a given delivery point at the same time, different categories of delivery points are introduced, etc). Additionally, the hybrid approach is planned to be complemented with fuzzy logic [15] and to be applied to project management [16] and complex systems [17].

The integration of CLP with metaheuristics such as GA (Genetic Algorithms) and ACO (Ant Colony Optimization) within the hybrid approach is also planned to enable solving problems of industrial size, which would be difficult to solve using the CLP/MP hybrid approach. It is also considering development of models to take account product demand interdependencies [18]. In the future it is planned to integrate proposed model with ERP and APS systems [19] and as a cloud internet application [20].

6 Reference

1. Russo F., Comi A., Polimeni A. Urban freight transport and logistics: Retailer's choices. In: Innovations in City Logistics (E. Taniguchi And R. G. Thompson eds.), Nova Science Publishers, Hauppauge Ny (USA), 2008, ISBN 978-1-60456-725-0
2. DG MOVE European Commission: Study on Urban Freight Transport FINAL REPORT By MDS Transmodal Limited in association with Centro di ricerca per il Trasporto e la Logistica (CTL), 2012.
3. Sitek, P., Wikarek, J. A Hybrid Approach to the Optimization of Multiechelon Systems, Mathematical Problems in Engineering, Article ID 925675, Hindawi Publishing Corporation, 2014, DOI:10.1155/2014/925675.

4. Sitek P., Wikarek J. A Hybrid Programming Framework for Modeling and Solving Constraint Satisfaction and Optimization Problems. Scientific Programming, vol. 2016, Article ID 5102616, 2016. doi:10.1155/2016/5102616.
5. Schrijver, A. Theory of Linear and Integer Programming. John Wiley &Sons, New York, NY, USA, 1998.
6. Apt, K., Wallace, M. Constraint Logic Programming using Eclipse. Cambridge: Cambridge University Press, 2006.
7. Rossi, F., Van Beek, P., Walsh, T. Handbook of Constraint Programming (Foundations of Artificial Intelligence). New York: Elsevier Science Inc, 2006.
8. Bocewicz G., Nielsen I., Banaszak Z. Iterative multimodal processes scheduling. Annual Reviews in Control, 38(1), 2014, 113-132.
9. Wikarek, J. Implementation aspects of Hybrid Solution Framework (HSF). Recent Advances in Automation, Robotics and Measuring Techniques Advances in Intelligent Systems and Computing, 267, 2014, 317-328, DOI: 10.1007/978-3-319-05353-0_31
10. Sitek P. A hybrid approach to the two-echelon capacitated vehicle routing problem (2E-CVRP). Advances in Intelligent Systems and Computing, 267, 2014, 251–263, DOI: 10.1007/978-3-319-05353-0_25.
11. Hooker J. N. Logic, optimization, and constraint programming. INFORMS Journal on Computing, vol. 14, no. 4, 2002, 295–321.
12. Bockmayr, A., Kasper, T. A Framework for Combining CP and IP, Branch-and-Infer, Constraint and Integer Programming: Toward a Unified Methodology Operations Research/Computer Science Interfaces, 27, 2014, 59–87.
13. Milano, M., Wallace, M. Integrating Operations Research in Constraint Programming. Annals of Operations Research, 175(1), 2010, 37 – 76.
14. Seuring, S., Müller, M. From a Literature Review to a Conceptual Framework for Sustainable Supply Chain Management. Journal of Cleaner Production 16, 2008, 1699-1710.
15. Kłosowski G., Gola A., Świć A. Application of Fuzzy Logic in Assigning Workers to Production Tasks. Distributed Computing and Artificial Intelligence, 13th International Conference, AISC, Vol. 474, 2016, 505-513, DOI: 10.1007/978-3-319-40162-1_54.
16. Relich, M. Identifying Project Alternatives with the Use of Constraint Programming. Borzemski, L. et al. (eds.), Information Systems Architecture and Technology, Advances in Intelligent Systems and Computing, vol. 521, Springer, 2017, 3–13.
17. Grzybowska K., Łupicka A. Knowledge Acquisition in Complex Systems. Proceedings of the 2016 International Conference on Economics and Management Innovations, part of Advances in Computer Science Research, vol 57, Yue X.-G., Duarte N.J.R. (eds.), 2016, 262-266, DOI: 10.2991/icemi-16.2016.5
18. Nielsen P., Nielsen I., Steger-Jensen K. Analyzing and evaluating product demand interdependencies. Computers in Industry, 61 (9), 2010, 869-876,. doi:10.1016/j.compind.2010.07.012.
19. Krenczyk, D. Jagodzinski, J. ERP, APS and Simulation Systems Integration to Support Production Planning and Scheduling. Advances in Intelligent Systems and Computing, Vol. 368, Springer International Publishing, 2015, 451-46,
20. Bak S., Czarnecki R., Deniziak S., Synthesis of Real-Time Cloud Applications for Internet of Things. Turkish Journal of Electrical Engineering &Computer Sciences, 2013. doi:10.3906/elk-1302-178.

Artificial bee colony algorithms for two-sided assembly line worker assignment and balancing problem

Mukund Nilakantan Janardhanan [1*], Zixiang Li[2], Peter Nielsen[1], Qiuhua Tang[2]

[1]Department of Mechanical and Manufacturing Engineering, Aalborg University, Denmark

Email: {mnj, peter}@m-tech.aau.dk

[2]Industrial Engineering Department, Wuhan University of Science and Technology, Wuhan 430081, Hubei, China

Email: zixiangli@126.com, tangqiuhua@wust.edu.cn

Abstract. Worker assignment is a new type of problem in assembly line balancing problems, which typically occurs in sheltered work centers for the disabled. However, only a few contributions consider worker assignment in a two-sided assembly line. This research presents three variants of artificial bee colony algorithm to solve worker assignment and line balancing in two-sided assembly lines. The utilization of meta-heuristics is motivated by the NP-hard nature of the problem and the chosen methods utilize different operators for onlooker phase and scout phase. The proposed algorithms are tested on 156 cases generated from benchmark problems. A comparative study is conducted on the results obtained from the three proposed variants and other well-known metaheuristic algorithms, such as simulated annealing, particle swarm optimization and genetic algorithm. The computational study demonstrates that the proposed variants produce more promising results and are able to solve this new problem effectively in an acceptable computational time.

Keywords: Assembly line balancing; two-sided assembly line; worker assignment; artificial bee colony; metaheuristics

1. Introduction

Extensively industry is using assembly lines to assemble different types of products, where a set of tasks is allocated to workstations and each workstation is assigned with one or several workers. In any real application, the problem of worker assignment requires consideration due to the different skills of the workers. This situation has been studied for sheltered work centers for the disabled [1, 2], where disabled workers might need more times to operate certain tasks or even are incapable of operating some tasks. The current state of task allocation mechanisms followed in assembly line balancing problems presently ignores this situation. Worker assignment and task allocation results in a new integrated assembly line worker assignment and balancing problem, which contains two interacted sub-problems. The worker assignment problem determines the

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_2

assignment of the workers on workstations and the assembly line balancing problem allocates tasks to workstations while satisfying different constraints.

Two-sided assembly lines are widely applied in automobile industries to assemble large-size high-volume products. In this type of line, workers operate tasks on two-faced workstations, referred to as mated-stations, in parallel [3, 4]. In this research worker assignment and balancing of a two-sided assembly line are taken into account simultaneously, a new integrated two-sided assembly line worker assignment and balancing problem (TALWABP) is proposed.

Even though there are plenty of contributions regarding two-sided assembly line balancing problems (TALBP) or worker assignment in one-sided assembly line, to the author's best knowledge there is no reported research on TALWABP where cycle time minimization is considered. Hence, this paper presents a first approach to solve the TALWABP with the objective of minimizing the cycle time. As a common objective function, the cycle time minimization criterion has great applications for the reconfiguration of the installed assembly lines[5]. A simple balancing problem in an assembly line is classified as NP-hard[4]; the considered problem also falls under this category due to the additional complexity incorporated. Hence, there is a need to use optimization techniques such as constraint programming and metaheuristics to solve large problem instances [1, 6, 7]. In this paper, several variants of the artificial bee colony (ABC) algorithm are developed to tackle the problem. A set of benchmarks are generated based on the benchmarks available in [8]. A comparative study on the proposed algorithms and three other well-known metaheuristics is presented and discussed in detail to demonstrate the performance of the proposed algorithms.

The reminder of the paper is organized as follows. Section 2 presents the problem description. Section 3 presents the details of the proposed algorithms. The comparative study and the statistical analysis are presented in Section 4. Finally, Section 5 concludes this research and suggests several future directions.

2. Problem definition

To scope the problem, we consider a set of assumptions as follows:

1) Workers have different skills and the operation times of tasks depend on the assigned workers.

2) All workstation is assigned a worker and the worker number is equal to the workstation number.

3) Some workers are incapable of operating some tasks and the corresponding operation times are set to a very large number when this situation happens.

In TALWABP, there is a special constraint on the preferred directions of tasks, referred to as direction constraint. Direction constraints can be grouped into three general types: L-type tasks, R-type tasks and E-type tasks. L-type tasks are allocated to the left side, R-type tasks are allocated to the right side and E-type tasks can be allocated to the left or right side[9]. In addition, there is a special condition for tasks on a mated-station, which is the existence of sequence-depended idle time[5]. Sequence-depended idle time is due to the precedence constraint and utilization of two sides, and can be reduced by

optimizing the task sequence on each workstation. Taking the sequence-depended idle time into account, our solution to the TALWABP optimizes not only the worker assignment and task allocation to workstations, but also the task sequence on each workstation. An example of worker assignment and task allocation on two-sided assembly line is depicted in Fig.1. In this figure, two facing workstations on the left side and right side comprise a mated-station. For instance, workstation (1, 1) and workstation (1, 2) comprise mated-station 1. Each workstation is assigned a worker and there are four workers on the four workstations. Two workers on a mated-station operate the allocated tasks simultaneously. Two-sided assembly line balancing determines the detailed allocation of tasks on each mated-station while worker assignment assigns the best-fit worker to each workstation.

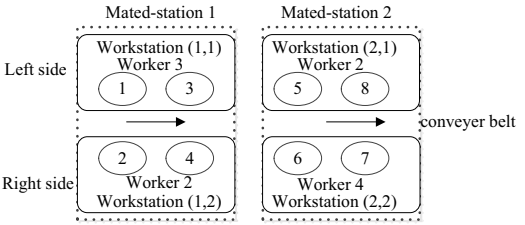


Fig.1 Worker assignment and task allocation on a two-sided assembly line

3. Proposed Methodology

Artificial Bee Colony (ABC) is one of the recently developed metaheuristic algorithm which has shown promising results for solving two sided assembly line balancing problem[5, 10] due to the faster convergence rate and there are only few parameters to be fine-tuned. This section first describes the procedure of the basic ABC algorithm, subsequently describes the proposed encoding and decoding policies along with the details of several variants of ABC algorithm.

3.1 ABC procedure

When solving optimization problems utilizing ABC algorithms, each solution is regarded as a food source and the fitness of an individual is referred to as the nectar amount. A pseudocode example of the ABC algorithm is presented in Fig. 2. Three groups of bees in the swarm search for the best food source, namely employed bees, onlookers and scouts. Employed bees exploit the nectar sources and provide the food sources' information to onlookers. Onlooker select food sources to exploit which emphasize intensification. The scout carries a random search to achieve a new food source to emphasize exploration. The procedure of the ABC algorithm is clarified in the following, where PS is the number of employed bees or onlookers. It should be noted that the number of employed bee is equal to onlookers and an onlooker becomes a scout.

Begin:
Initialize the swarm with *PS* individuals; *% Initialization*
While (termination criterion is satisfied**) do**
For *p*=1 to *PS* **do** *% Employed bee*
Achieve a new individual based on the incumbent individual *p* utilizing neighbor operator;
Replace the incumbent one with new one when better fitness value is achieved;
For *p*=1 to *PS* **do** *% Onlooker*
Select an individual based on probability value for each food source;
Achieve a new individual based on the selected individual utilizing neighbor operator;
Replace the incumbent one with new one when better fitness value is achieved;
If (One individual should be abandoned**)** *% Scout*
Determine the abandoned individual;
Replace the abandoned individual with the new one achieved by a scout
Endif
End

Fig.2 Pseudocode of ABC algorithm

3.2 Encoding and decoding

Since the addressed problem is similar to the robotic TALBP in [11] this paper proposes a similar encoding method which utilizes two vectors: task permutation and worker assignment. It is to be noted that the task permutation vector is selected since it has shown superior performance than the mated-station oriented encoding [5]. In the two vectors, the worker assignment vector determines the detailed worker assignment and task permutation determines the sequence of allocating tasks. The former task in the task permutation is allocated first. An example with 12 tasks and 2 mated-station is illustrated in Fig. 3 to describe the solution presentation. It is observed that the worker assignment vector is 3, 2, 1 and 4, and thus worker 3 and 2 are assigned to the left side and right side of mated-station 1 and worker 1 and 4 are assigned to the left side and right side of mated-station 2. The three former tasks in the task permutation are task 3, 2 and 6. Hence, task 3, 2 and 6 are allocated first and are all allocated to mated-station 1.

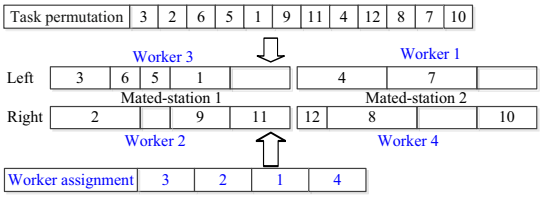


Fig.3 Illustrated solution presentation

However, the task permutation is not a feasible task allocation, and an initial cycle time and an effective decoding scheme are necessary to transfer the task permutation

into a feasible task allocation. Based on the task permutation, the decoding procedure in Li et al. [11] is proposes to transfer the task permutation into a feasible task allocation. In this decoding procedure, tasks are allocated to former workstations as much as possible gradually, and the last mated-station endures all the unallocated tasks. The largest finishing time of all mated-stations is regarded with the cycle time achieved by this task permutation. Regarding the initial cycle time (CT), it is set to be a large number in this paper. The initial cycle time is updated with $\mathrm{CT} \leftarrow \mathrm{CT}_N - 1$ when a smaller new best cycle time CT_N is achieved or $\mathrm{CT} \leftarrow \mathrm{CT} - 1$ when the current best cycle time remains unchanged. The above expression guarantees that the initial cycle time reduces over the algorithm's evolution, and as a consequence, the achieved best cycle time is reduced. When new best cycle time is achieved, all the individuals are re-decoded using the initial cycle time of $\mathrm{CT}_N - 1$ and the new achieved cycle times are set to their objective functions. This method, referred to as the iteration mechanism, ensures that all the individuals are evaluated using the same initial cycle time.

3.3 Proposed ABC methods

This paper develops three variants of ABC algorithms, which are different in onlooker phase and scout phase. The first variant, referred to as ABC1, is the traditional ABC. The other two versions, referred to as ABC2 and ABC3, have some modifications. The features of the three method are listed as follows.

ABC1: In the onlooker phase, the selection probability value for food source p, Pr_p is calculated with $\mathrm{Pr}_p = \left(1/CT_p\right)/\sum_{p'}^{PS}\left(1/CT_{p'}\right)$, where CT_p is the cycle time achieved by individual p. Then an individual is selected using roulette selection. In the scout phase, if the best cycle time is not updated, the duplicated or worst individual with the largest cycle time is replaced with a new randomly generated solution. **ABC2:** ABC2 differs from ABC1 in the way the duplicated or worst individual with the largest cycle time is replaced with a neighbor solution of the randomly selected individual. **ABC3:** ABC3 differs from ABC 1 in both the onlooker and scout phase. In the onlooker phase, ABC3 utilize tournament selection, where two individuals are randomly selected and the individual with the smaller cycle time is finally selected. In the scout phase, the duplicated or worst individual with the largest cycle time is replaced with a neighbor solution of the randomly selected individual.

Compared with the roulette selection in the onlooker phase, tournament selection is much simpler. In the scout phase, the randomly generated solution always has poor performance. Thus, ABC2 and ABC3 utilize the neighbor solution of the randomly selected individual to replace the abandoned one. Apart from the above factor, this neighbor operator is another important factor and all the algorithms share the same neighbor operator. In this neighbor operator, this research employs both insert operation and swap operation to modify either the task permutation vector or worker assignment vector. The insert operation and swap operation have been widely applied in assembly line balancing problems, such as robotic TALBP in Li et al. [11]. Since two vectors are involved, the approach in this research is to first select a vector randomly and later select insert operation or swap operation randomly to modify the selected vector.

4. Computational Study

This section presents the details of the tests conducted on the proposed method and their findings. The utilized benchmark problems are first introduced and the comparative study is presented later. Since there is no available benchmark problem for this new TALWABP, this research generates a set of tested problems on the basis of the original TALBP benchmarks summarized in Tang et al. [5]and Li et al. [9]. The operation times of tasks are produced using the method in Chaves et al. [8], where two variabilities of the operation times are employed: low variability and high variabilities. The first workers have the same operation times published in literature, and the operation times for the remaining workers are generated as follows. Regarding low variability, the operation times of task *i* by workers are randomly generated within a uniform distribution of [1, t_i] where is t_i the original operation times of task *i* in literature. Regarding high variability, the operation times of task *i* by workers are randomly generated within a uniform distribution of [1,3×t_i]. In addition, two ratios of task-worker incompatibilities are utilized: low ratio with 10% and high ratio with 20%. Since the original benchmarks contain 39 cases [12], and there are 156 combinations due to two operation time variabilities and two incompatible percentages. This new benchmark set is summarized in Table 1, where *Nt* is the number of tasks and *Nm* is the number of mated-stations.

Table 1 Description of the tested benchmarks

Problem	*Nt*	*Nm*	Time variabilities	Incompatibilities
P9	9	2,3	Low, high	Low, high
P12	12	2,3,4,5	Low, high	Low, high
P16	16	2,3,4,5	Low, high	Low, high
P24	24	2,3,4,5	Low, high	Low, high
P65	65	4,5,6,7,8	Low, high	Low, high
P148	148	4,5,6,7,8,9,10,11,12	Low, high	Low, high
P205	205	4,5,6,7,8,9,10,11,12,13,14	Low, high	Low, high

4.1 Computational evaluation

This section presents the comparative study and three other algorithms are adopted: simulated annealing algorithm[12], particle swarm optimization algorithm [11] and genetic algorithm[13]. All the tested algorithms are calibrated using full factor design following Tang et al. [5] and [11]. All cases are solved by all the combinations of the parameters and they terminate when CPU time reaches $\mathrm{Nt} \times \mathrm{Nt} \times \tau$ milliseconds, where τ is set to 10. After executing all the experiments, the relative percentage deviation (RPD) is proposed to transfer the achieved cycle times using $\mathrm{RPD} = 100 \times (CT_{some} - CT_{Best})/CT_{Best}$, where CT_{some} is the cycle time yield by one combination and CT_{Best} is the cycle time achieved by all the combinations. The multifactor analysis of variance (ANOVA) technique is applied to analyze the achieved RPD values as done in Li et al. [11]. Based on the determined parameters, all algorithms are solved using 156-benchmark problems under the two-termination criterion of $Nt \times Nt \times 10$ milliseconds and $Nt \times Nt \times 20$ milliseconds. Table 2 presents the average RPD values of tested algorithms, where the average RPD value in each cell is the average value of several cases. For instance, the value for P205 is the average value of 11

cases. It can be observed that ABC3 is the best performer with an average RPD value of 3.90 under the first termination criterion. ABC2 is the second best performer with an average RPD value of 4.15. PSO and GA algorithms showed worst performance when compared. When the computational time reaches $Nt \times Nt \times 20$ milliseconds, ABC3 still ranks the first and ABC ranks the second according to the increasing order of the average RPD values. Among the remaining algorithms, SA ranks third, ABC1 ranks fourth, GA ranks fifth and finally PSO ranks sixth. In addition, ABC3 is the best performer among the three ABC variant for P24 and P148 using the second termination criterion. ABC3 is the best performer for P65 and P205. These computational results suggest that the ABC2 and ABC3 are quite effective when solving TALWABP due to its ability to achieve good solution in short computation time.

Table 2 Average RPD values of six tested algorithms

Problem	Tested cases	Average relative percentage deviation						CPU time(s)
		SA	PSO	GA	ABC1	ABC2	ABC3	
Nt × Nt × 10								
P9	2	0.00	0.00	0.00	0.00	0.00	0.00	0.81
P12	4	0.00	2.08	1.56	0.00	1.56	0.00	1.44
P16	4	0.42	0.00	0.00	0.00	0.00	0.00	2.56
P24	4	0.48	4.01	2.09	0.00	0.42	0.51	5.76
P65	5	6.05	32.54	11.40	7.28	4.90	5.09	42.25
P148	9	11.43	39.78	16.04	15.78	7.40	6.88	219.04
P205	11	8.45	43.16	13.64	13.35	5.70	5.71	420.25
Average RPD		5.89	26.15	9.39	8.34	4.15	**3.90**	-
Nt × Nt × 20								
P9	2	0	0.00	0.00	0.00	0.00	0.00	1.62
P12	4	0.00	2.08	1.56	0.00	0.00	0.00	2.88
P16	4	0.42	0.00	0.00	0.00	0.00	0.00	5.12
P24	4	0.48	2.93	1.11	0.00	0.42	0.15	11.52
P65	5	5.54	28.97	10.12	4.29	3.49	3.66	84.50
P148	9	8.01	36.57	14.17	11.74	3.69	2.75	438.08
P205	11	5.16	40.35	11.47	9.10	2.64	3.24	840.50
Average RPD		4.10	24.05	8.08	5.83	2.09	**2.03**	-

5. Conclusion and future research

Worker assignment is a new problem in assembly line balancing problems, which arises due to different skills of the worker. This research considers the integrated worker assignment and balancing problem in two-sided assembly line systems with the objective of optimizing cycle time. Three variants of the artificial bee colony (ABC) algorithm are developed to solve this new problem, where the first variant is the original ABC algorithm and two latter variants are improved editions. Since no benchmark problems are available, a total number of 156 cases are generated ranging from P9 with 9 tasks to P205 with 205 tasks. To evaluate the performance of the proposed algorithms, three other algorithms are implemented and compared. Computational results show that the two improved ABC algorithms are the two best performers and produce promising results for this new problem. Future researchers might consider more constraints in real application, such as positional constraint and zoning constraint. It is also interesting to

develop more advanced algorithm such co-evolutionary algorithms to solve the problem.

References

1. Miralles, C., Garcia-Sabater, J.P., Andres, C., Cardos, M.: Advantages of assembly lines in sheltered work centres for disabled. A case study. International Journal of Production Economics 110, 187-197 (2007)
2. Miralles, C., García-Sabater, J.P., Andrés, C., Cardós, M.: Branch and bound procedures for solving the assembly line worker assignment and balancing problem: Application to sheltered work centres for disabled. Discrete Applied Mathematics 156, 352-367 (2008)
3. Bartholdi, J.: Balancing two-sided assembly lines: a case study. The International Journal Of Production Research 31, 2447-2461 (1993)
4. Li, Z., Tang, Q., Zhang, L.: Two-sided assembly line balancing problem of type I: Improvements, a simple algorithm and a comprehensive study. Computers & Operations Research 79, 78-93 (2017)
5. Tang, Q., Li, Z., Zhang, L.: An effective discrete artificial bee colony algorithm with idle time reduction techniques for two-sided assembly line balancing problem of type-II. Computers & Industrial Engineering 97, 146-156 (2016)
6. Relich, M., Świc, A., Gola, A.: A knowledge-based approach to product concept screening. In: Distributed Computing and Artificial Intelligence, 12th International Conference, pp. 341-348. Springer, (Year)
7. Relich, M., Bzdyra, K.: Knowledge discovery in enterprise databases for forecasting new product success. In: International Conference on Intelligent Data Engineering and Automated Learning, pp. 121-129. Springer, (Year)
8. Chaves, A.A., Miralles, C., Lorena, L.A.N.: Clustering search approach for the assembly line worker assignment and balancing problem. In: Proceedings of the 37th international conference on computers and industrial engineering, Alexandria, Egypt, pp. 1469-1478. (Year)
9. Li, Z., Tang, Q., Zhang, L.: Minimizing the cycle time in two-sided assembly lines with assignment restrictions: improvements and a simple algorithm. Mathematical Problems in Engineering (Article ID 4536426) :1-15,
10. Tapkan, P., Ozbakir, L., Baykasoglu, A.: Modeling and solving constrained two-sided assembly line balancing problem via bee algorithms. Applied Soft Computing 12, 3343-3355 (2012)
11. Li, Z., Janardhanan, M.N., Tang, Q., Nielsen, P.: Co-evolutionary particle swarm optimization algorithm for two-sided robotic assembly line balancing problem. Advances in Mechanical Engineering 8, 1687814016667907 (2016)
12. Özcan, U., Toklu, B.: Balancing of mixed-model two-sided assembly lines. Computers & Industrial Engineering 57, 217-227 (2009)
13. Kim, Y.K., Song, W.S., Kim, J.H.: A mathematical model and a genetic algorithm for two-sided assembly line balancing. Computers & Operations Research 36, 853-865 (2009)

Cyclic steady state behavior subject to grid-like network constraints

Bocewicz G.[1], Wójcik R.[2], Banaszak Z.[1]

[1] Faculty of Electronics and Computer Science, Koszalin University of Technology, Poland, (e-mail: bocewicz@ie.tu.koszalin.pl).
[2] Department of Computer Engineering, Faculty of Electronics, Wrocław University of Science and Technology, Poland (e-mail: robert.wojcik@pwr.edu.pl)

Abstract. The paper's objective concerns assessing a structure constraints of a grid-like network of periodically acting local cyclic processes interacting via shared resources from the perspective of Cyclic steady States Space (CSS) reachability. Interaction of concurrently flowing local processes follows a mutual exclusion protocol determined by a set of priority dispatching rules determining an order in which competing processes can access to common shared resources. The main problem is in essence to identify conditions deciding about the scale and amount of cyclic steady states reachable in such networks. Such conditions allows one to replace the exhaustive search for the admissible deadlock-free control of the whole network by focusing on its small size parts. The proposed methodology behind fast assessment of cyclic steady states reachable in grid-like structure of cyclic process is clarified through multiple illustrative examples.

Keywords: Periodic Vehicle Routing Problem, re-scheduling, fleet of AGVs, multimodal process, declarative modelling, cyclic scheduling.

1 INTRODUCTION

The increasing trend among commercial firms for on-line operations and the consequent need by active users for responsive systems as well as distributed processing exhibiting a high degree of resource and data sharing, cause of situations in which deadlock is a constant threat. Deadlocks arise as a consequence of exclusive access and circular wait, i.e. when members of a group of processes which hold resources are blocked indefinitely from access to resources held by other processes within the group.

Guarantee of the smooth, i.e. deadlock-free, execution of concurrently flowing processes of different nature and character, e.g. traffic, production, and data/communication flows, plays a crucial role in course of their control design. Numerous papers have been published to address deadlock handling strategies [1, 5-11]. Usually employed are based on one of the following two schemes: prevention which is aimed at constraining system users so that requests leading to deadlock never occur, and an avoidance which aims to guarantee a resource request only if at least one way

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_3

remains for all processes to complete execution [10]. A prevention mechanism differs from an avoidance scheme (which is NP-complete [13]) in that the system need not perform run-time testing of potential allocations.

Since deadlock-freeness of concurrently executed processes implies that at least one cyclic steady state of their flow there exists, hence instead of searching for the mechanisms guaranteeing deadlock-free execution one may concentrate on conditions guaranteeing reachability of a cyclic steady state [2,3,4]. Due to this idea, our contribution is devoted to methodology supporting assessment of reachability of possible cyclic steady states in a given network following assessment of steady states reachable in its substructure. An idea standing behind this approach assumes that by solving a small-scale computationally hard problem (associated with a part of a grid-like network), one can solve, online, a large scale problem associated with a whole network.

The remainder of the paper is organized as follows: Section 2 providing a state graph model of grid-like structure of concurrently flowing cyclic processes introduces to the concept of cyclic steady state and formulates problem statement. Section 3 elaborates on steady state reachability conditions and supporting them research methodology as well as on constraint satisfaction problem. Section 4 provides examples illustrating relationship linking a way of extraction of a sample-like subnetwork on assessment of Cyclic Steady State Space (CSS for short) reachable in a whole network. Concluding remarks and further work are submitted in Section 5.

2 CONCURRENTLY FLOWING CYCLIC PROCESSES

The relationships between a set of processes and a set of distinct resources in use by these processes can be described by a state graph [3].

2.1 Grid-like structure modeling

To illustrate the approach proposed, let us consider the network composed of concurrently flowing cyclic processes shown in Fig.1. Each elementary process see Fig. 1 b) consists of 6 resources shared with other conterminous processes. Let us assume operation times executed in each process are the same and equal to 1 t.u. (time unit). So, each separately treated elementary process can be seen as periodic with the cycle time equal to 6 t.u. Due to assumed state graph model the systems of concurrently flowing cyclic processes P [3,4] can be modeled by digraphs whose nodes correspond to the set of resources R and whose arcs are defined so that: if some process P_i has access to resource R_1 and is waiting for access to resource R_2, then there exists an arc directed from node R_1 to node R_2, see Fig.1 b). In further considerations arcs' arrows are omitted, however the direction of processes execution ermines distinguished by twisted arrows. In the network considered three samples of Elementary Pattern Networks (EPNs) are distinguished: EPN_1, EPN_2 and EPN_1. The EPNs selected do not exhaust all possible cases of EPNs which may differ in shape and in scale (i.e. in the number of elementary structures they contain). However, their common feature is that they can be tessellated (like a mosaic) to form a pattern that recreates a given regular grid-like structure.

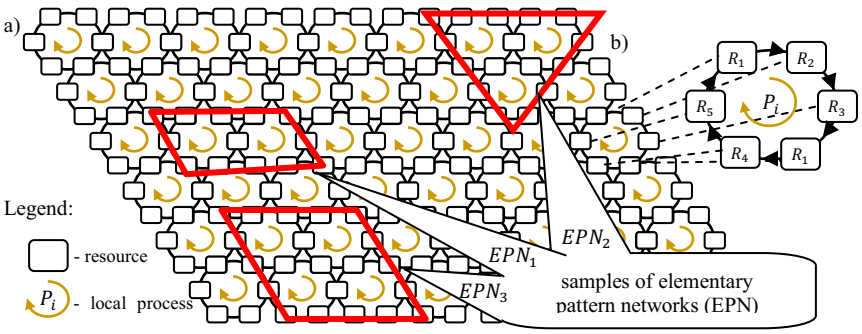


Fig. 1. Structure of exemplary grid-like network distinguishing EPN_1, EPN_2 and EPN_3 a), and the state graph model of local cyclic process b)

2.2 Cyclic steady state behavior

Consider a set of priority dispatching rules implementing the process synchronization protocol guaranteeing non-preemptive scheduling of resources and following mutual exclusion condition. Let us assume that each c-th priority dispatching rule σ_c (allocated to the c-th resource) settles resource conflicts related to the order in which processes/streams can access shared resources R_c (e.g. $\sigma_3 = (P_1, P_2)$ from Fig. 1b) is a sequence which determines the order of processes supported by resource R_3: $\ldots, P_1, P_2, P_1, P_2, P_1, P_2, \ldots$).

Let $\mathbb{S}$ stand for a set of admissible system states, i.e. states which specify a way of allocation of processes to resources. Moreover, let set $\mathbb{S}' \subseteq \mathbb{S}$ is a set of states reachable from the initial state $S^0 \in \mathbb{S}'$, i.e. belonging to the sequence of states $(S^0, S^h, \ldots, S^g, S^d, S^i, S^k, \ldots, S^w, S^r)$ wherein each successive state is reachable, in accordance with Θ, from a state immediately preceding it. In case the state S^0 is reachable from the stateS^r being the last in the sequence, the resulting cyclic steady state behavior can be defined as a triple:

$$B = (\Theta, S^0, \mathbb{S}'), \tag{1}$$

where: $\Theta = \{\sigma_c = (s_{c,1}, \ldots, s_{c,d}, \ldots, s_{c,lh(c)}) \,|c = 1 \ldots lk\}$ – a set of dispatching rules, where $s_{c,d} \in P$ – the d-th process sharing resource R_c ; $\mathbb{S}' = \{S^r | \, S^r = (A^r, Z^r)$, S^r is reachable from S^0 in accordance with $\Theta, S^r \in \mathbb{S}\}$; S^0 – the initial state of the state graph model (initial allocation of processes); $S^r = ({a_1}^r, {a_2}^r, \ldots, {a_c}^r, \ldots, {a_{lc}}^r)$– the r-th state of the state graph model, which determines allocation of local processes, ${a_c}^r \in P \cup \{\Delta\}$, ${a_k}^r = P_i$ – an allocation of a process which means that the c-th resource R_c is occupied by process P_i, and ${a_c}^r = \Delta$ – means that the c-th resource R_c is free.

In case of EPN_1, the cyclic steady state reachable from the pair $(\Theta = \{\sigma_1 = (P_1); \sigma_2 = (P_2, P_1); \sigma_3 = (P_2, P_1); \sigma_4 = (P_2, P_1); \sigma_5 = (P_2, P_1); \sigma_6 = (P_2)\}, S^0 = (P_1, \Delta, \Delta, \Delta, \Delta, P_2))$, consists of states, $\mathbb{S}' = \{S^1 = (\Delta, P_1, \Delta, \Delta, \Delta, P_2); S^2 = (\Delta, P_1, \Delta, \Delta, \Delta, P_2); \ldots . S^{10} = (P_1, \Delta, P_2, \Delta, \Delta, \Delta)\}$), and its graphical representation expressed by the Gantt's chart of cyclic schedule is shown in Fig. 2a).

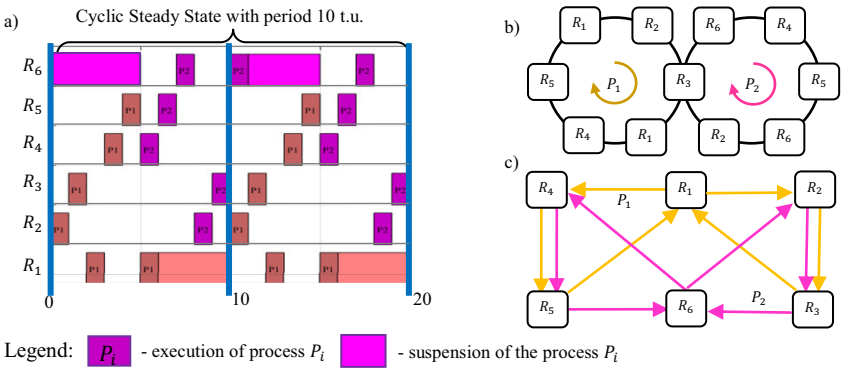


Fig. 2. Cyclic schedule following $FEPN_1$ a), the state graph models of EPN_1 b), and $FEPN_1$ c)

2.3 Problem statement

In the light of the above considerations, a question arises whether there exists an ordered pair: ***(initial process allocation, a set of dispatching rules)*** encompassing cyclic steady state of the network. The problem of whether such a pair exists is a computationally hard problem, which limits the analysis to small-scale cases, much smaller than those encountered in practice.

It can be shown, however, that the behavior of a given network can be predicted on the basis of the behavior of its EPNs [2]. It is easy to observe that an initial allocation of local processes in a regular network will be followed in each individual EPN by subsequent allocation of local processes in compliance with the same priority dispatching rules. That is because if one replicates the same initial process allocation in all the remaining EPNs, the structure will be free of collisions between processes which use the same resources, i.e. deadlock-free. This means that by solving a small-scale computationally hard problem (associated with an EPN), one can easy solve a large scale problem associated with a whole network.

Moreover, since different conflict resolutions of processes executed in a EPN results in different periods α of its cyclic steady state, different steady states of the whole network can also be prototyped. In that context, our steady state reachability problem can be reduced to the question of whether it is possible to extract the EPN guaranteeing its steady state follows values of assumed performance indices.

3 STEADY STATE REACHABILITY

Consider a class of state graph models of grid-like networks represented by connected digraphs comprised of elementary substructures, e.g. the EPNs, such that:

– each EPN is composed of elementary cyclic digraphs modeling local cyclic processes,

– the EPNs can be used to tessellate a given grid-like network,
– each EPN has a corresponding Folded-form of EPN (FEPN – for short) which is formed by gluing together the vertices of the EPN which in network are joined with the corresponding fragments of the adjacent EPNs.

An example of the EPN and its FEPN is shown in Fig. 2c). In general case one may consider different cases of EPNs which may differ in shape and in scale, however, they can be tessellated to form a pattern that recreates a given grid-like network.

3.1 Research methodology

It can be shown [4] that cyclic steady state, characterized by the behavior B described by fixed periodic execution of local cyclic processes, occurs when:

- it has a finite number of states $\mathbb{S}'$,
- $\forall S^i \in \mathbb{S}'$, state S^i is reachable from S^{i-1} and state S^0 is reachable from $S^q \in \mathbb{S}'$ in accordance with the dispatching rules adopted.

Moreover, due to [2] a steady state of the FEPN following EPN extracted from a given network entails a cyclic steady state of the whole grid-like network. That means, the CSS of a given network can be predicted on the basis of the cyclic behaviour of its EPNs, specified by both: a set of dispatching rules Θ_{FEPN}, and an initial state $S^0{}_{\text{FEPN}}$.

In that context our cyclic steady state reachability problem can be reduced to the question of whether it is possible to extract the EPN such that there exists a pair (Θ_{FEPN}, $S^0{}_{\text{FEPN}}$) guaranteeing cyclic steady state of FEPN and consequently a cyclic behaviour of the whole grid-like network?

The graphical illustration of the research methodology used for this study consisting of the following four stages, is shown in Fig. 3.

1. Identify a set of potential EPNs that could be used to tessellate a given grid-like network
2. Consider a set of convoluted representations FEPNs corresponding to identified EPNs
3. Searching for an initial state S^0_{FEPN} and a set of dispatching rules Θ_{FEPN} allocation of which results in cyclic schedule encompassing behavior of considered FEPN,
4. Assessment of behavior of whole grid-like network as implied by considered FEPN.

Summing up, since a cyclic behavior of a folded form of the EPN entails a cyclic behavior of the grid-like network, hence solving a small-scale computationally hard problem at low time-cost, a large scale problem associated with a whole network can be solved in online mode.

Moreover, because different conflict resolutions of processes executed in a FEPN results in different periods α of its cyclic steady state, different behaviors of the whole grid-like networks can also be assessed. Due to above methodology each EPN's steady state is searched in the context of the relevant FEPN.

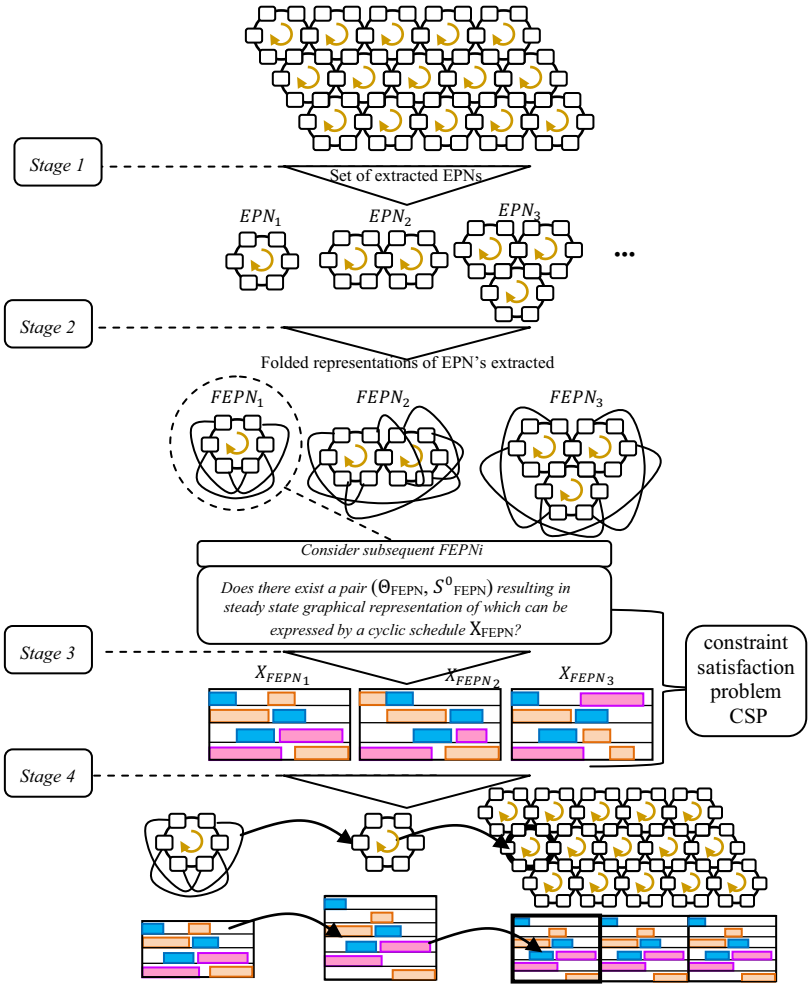


Fig. 3. Methodology of CSS searching

3.2. Constraint satisfaction problem

In order to determine Θ_{FEPN}, S^0_{FEPN}, guaranteeing the attainability of the cyclic schedule X_{FEPN} within the FEPN, the following constraint satisfaction problem can be considered (2):

$$CS_{FEPN} = \left(\left(\{X_{FEPN}, \Theta_{FEPN}, S^0_{FEPN}, \alpha_{FEPN}\}, \{D_X, D_\Theta, D_S, D_\alpha\}\right), C\right), \quad (2)$$

where:

$X_{FEPN}, \Theta_{FEPN}, S^0_{FEPN}, \alpha_{FEPN}$ – decision variables, X_{FEPN} – the cyclic schedule of substructure FEPN, Θ_{FEPN} – dispatching rules of FEPN, S^0_{FEPN} – the initial state, α_{FEPN} – the period of processes for FEPN,

$D_X, D_\Theta, D_S, D_\alpha$ – domains determining admissible values of decision variables: D_X:, $x_{i,j} \in \mathbb{Z}$; D_S: $S^0_{\mathrm{FEPN}} \in \mathbb{S}'$, D_α: $\alpha_{FEPN} \in \mathbb{Z}$,

C – the set of constraints describing grid-like network behavior, the set C includes the constraints describing the execution of local cyclic processes [2,3,4].

The schedule X_{FEPN} and $\Theta_{\mathrm{FEPN}}, S^0_{\mathrm{FEPN}}$ that meets all the constraints from the given set C is the solution sought for the problem (2). It can be obtained using standard constraint programming driven software platforms such as ILOG, OzMozart, ECLiPSE [12].

4 CYCLIC STEADY STATE SPACE CSS

The main objective of this section is to provide some observations useful in the course of cyclic steady states generation subject to constraints imposed by grid-like networks. The rules enabling to estimate the period of the steady state cycle time imposed by cyclic processes structure and the dispatching rules employed are the main goal.

4.1 Cyclic steady states reachability

As it was already mentioned a CSS consists of family of sets $\mathbb{S}'$ such that:

- $\forall S^i \in \mathbb{S}', i \in \{1,2,\dots,n\}$, state S^i is reachable from S^{i-1} and state S^0 is reachable from $S^n \in \mathbb{S}'$ or
- there are subsets $\mathbb{S}^*, \mathbb{S}^{**} \subset \mathbb{S}'$ such that $\mathbb{S}^* \cap \mathbb{S}^{**} = \varnothing$ and $\mathbb{S}^* \cup \mathbb{S}^{**} = \mathbb{S}'$ and $\forall S^i \in \mathbb{S}^{**} i \in \{2,\dots,n'\}$, state S^i is reachable from S^{i-1} and state S^1 is reachable from $S^{n'} \in \mathbb{S}'$, and $\exists\, S^{0k},\dots,S^{0l} \in \mathbb{S}^*$ which are not reachable from any state from the remaining ones and such that $\forall S^i \in \mathbb{S}^{**}$ is reachable from all of them.

Example of the CSS imposed by FEPN_2 from Fig. 2 c) is illustrated in Fig. 4. Problem (3) considered for this case has been implemented and solved in the constraint programming environment OzMozart (CPU Intel Core 2 Duo 3GHz RAM 4 GB).

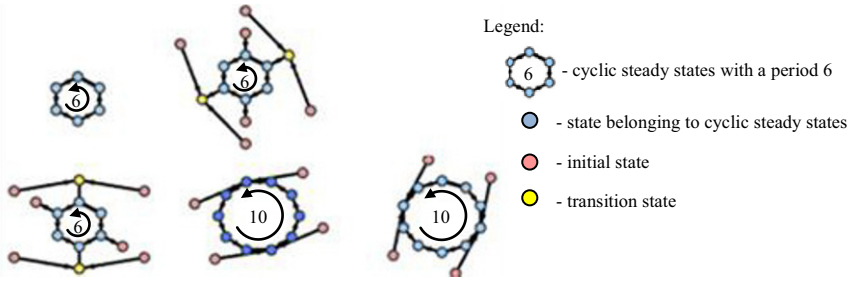
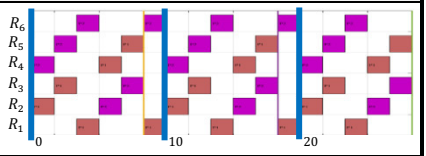
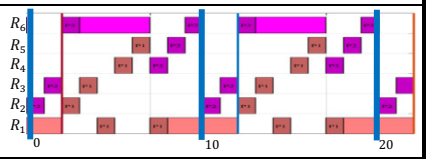


Fig. 4. CSS following FEPN_1 from Fig. 2 c).

More detailed assessments of steady states following EPN_1 structure are collected in Tab. 1. It should be noted, however, that not all grid-like networks have EPN resulting in any (S_{CEPN}, Θ_{FEPN}) [14].

Table 1. Periods and schedules implied by (S^0, Θ_1) and (S^0, Θ_2) for $FEPN_1$ from Fig. 2 c)

S^0	Dispatching rules θ	Period α	Schedule
$S^0 = (P_1, \Delta, \Delta, \Delta, \Delta, P_2)$	Θ: $\sigma_1 = (P_1)$; $\sigma_2 = (P_1, P_2)$; $\sigma_3 = (P_1, P_2)$; $\sigma_4 = (P_1, P_2)$; $\sigma_5 = (P_1, P_2)$; $\sigma_6 = (P_2)$	$6\ t.u.$	
	Θ: $\sigma_1 = (P_1)$; $\sigma_2 = (P_2, P_1)$; $\sigma_3 = (P_2, P_1)$; $\sigma_4 = (P_2, P_1)$; $\sigma_5 = (P_2, P_1)$; $\sigma_6 = (P_2)$	$10\ t.u.$	

Main findings resulting from above experiments include the following observations:

- there are grid-like networks such that corresponding to them CSSs are empty sets,
- the different cyclic steady states $\mathbb{S}', \mathbb{S}''$ following the same EPN may include the same states, $\mathbb{S}' \cap \mathbb{S}'' \neq \varnothing$,
- if different cyclic steady states $\mathbb{S}', \mathbb{S}''$ shaped by Θ' and Θ" follow the same EPN, then there exists $S^i \in \mathbb{S}' \cap \mathbb{S}'' \neq \varnothing$, it means for the same initial state S^0 may exist different Θ', Θ" resulting in different cyclic steady states,
- any state from cyclic steady state can be assumed to be an initial state,
- searching for $(S_{CEPN}, \Theta_{\mathrm{CEPN}})$ can be seen as alternately executed searching for Θ' subject to a given S^0, and searching for S^0 subject to a given Θ' .

4.2 Cyclic steady state space following possible EPN extraction

Since there are many different EPNs that could be extracted from a given grid-like network, the main issue concerns the relationships linking cyclic steady states following different EPNs. So, in order to analyze selected quantitative and qualitative differences occurring between CSSs while caused by different ESPs let us consider EPN_1, EPN_2 and EPN_3 from Fig. 1 a). Folded versions of EPN_2 and EPN_3 are shown in Fig. 5 a) and Fig. 5 b), respectively. Detailed assessments of reachable CSSs are collected in Tab. 2.

Gantt's charts of cyclic schedules describing selected cyclic steady states following considered $FEPN_2$ and $FEPN_3$ are shown in Fig. 6 a) and Fig. 6 b), respectively.

Main findings resulting from above experiments include the following observations:

- the growing complexity (measured by a number of included local cyclic processes) of an EPN results in an exponential size grow of its CSS, caused by rapidly growing difference between resources' quantity and a number of utilizing them processes,
- if an EPN' which may differ in shape and in scale, however, can be tessellated to form a pattern that recreates a given EPN", then following them CSSs consist of cyclic steady states characterized by the same periods,

- values of minimal periods of cyclic steady states belonging to CSSs following possible SFPN are the same.

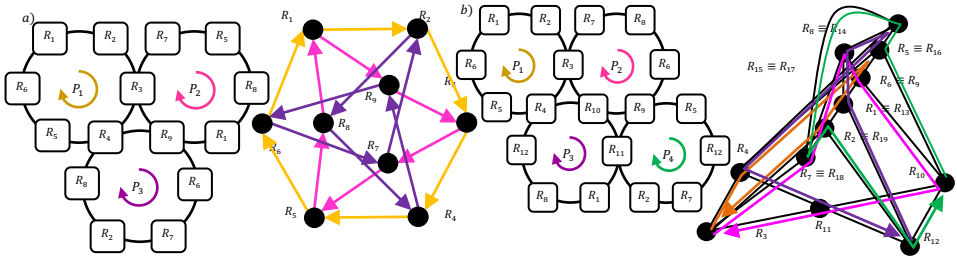


Fig. 5. EPN_2 and its folded version a), EPN_3 and its folded version b)

Table 2. Comparison of CSSs following $FEPN_1$ from Fig. 2 c) and $FEPN_2$, $FEPN_3$ from Fig. 5

FEPN	Cardinality of CSS	Period α	Number of cyclic steady states	Exemplary Initial State S^0	Exemplary Dispatching rules Θ	Computing time
$FEPN_1$	5	$6\ t.u.$	3	$(P_1, \Delta, \Delta, \Delta, \Delta, P_2)$	Θ: $\sigma_1 = (P_1)$; $\sigma_2 = (P_1, P_2)$; $\sigma_3 = (P_1, P_2)$; $\sigma_4 = (P_1, P_2)$; $\sigma_5 = (P_1, P_2)$; $\sigma_6 = (P_2)$	40 secs
		$10\ t.u.$	2			
$FEPN_2$	13	$6\ t.u.$	7	$(P_1, \Delta, \Delta, \Delta, P_2, \Delta, \Delta, \Delta, \Delta, P_2, \Delta)$	Θ: $\sigma_1 = (P_1, P_2)$; $\sigma_2 = (P_1, P_3)$; $\sigma_3 = (P_1, P_2)$; $\sigma_4 = (P_1, P_3)$; $\sigma_5 = (P_2, P_1)$; $\sigma_7 = (P_3, P_2)$; $\sigma_8 = (P_2, P_3)$; $\sigma_9 = (P_3, P_2)$;	24 hours
		$15\ t.u.$	6			
$FEPN_3$	25	$6\ t.u.$	16	$(P_1, \Delta, \Delta, \Delta, \Delta, \Delta, P_2, \Delta, \Delta, \Delta, \Delta, P_4, P_3)$	Θ: $\sigma_1 = (P_1, P_3)$; $\sigma_2 = (P_4, P_1)$; $\sigma_3 = (P_1, P_2)$; $\sigma_4 = (P_1, P_3)$; $\sigma_5 = (P_1, P_4)$; $\sigma_8 = (P_2, P_3)$; $\sigma_9 = (P_2, P_4)$; $\sigma_{10} = (P_2, P_3)$; $\sigma_{11} = (P_4, P_3)$; $\sigma_{12} = (P_3, P_4)$;	50 hours
		$10\ t.u.$	9			

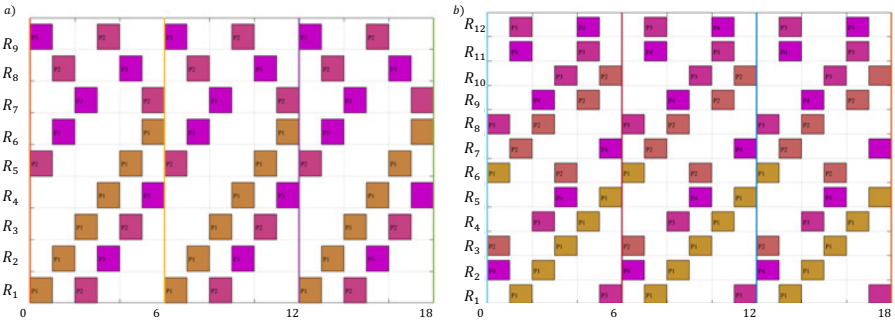


Fig. 6. Cyclic schedules depicting cyclic steady states following $FEPN_2$ a), and $FEPN_3$ b).

5 CONCLUSIONS

The majority of problems stated within a framework of grid-like networks can be seen as concerning its schedulability, and especially decidability of cyclic scheduling problems. Such problems are quite common in the systems with high density of processes (e.g. in transportation systems) where the question of attainability assessment of the cyclic steady state behavior becomes crucial. Since these kind of problems are NP-hard in nature, hence the feasible solutions can be obtained only for small scale instances. In that context a declarative approach to the reachability problem that can be employed by decision-makers in order to generate, analyze and evaluate cyclic steady states reachable in a given grid-like network has been proposed. Delivered solution employs observation that the CSS of a given network can be predicted on the basis of the CSS of its EPNs. That means that by solving a small-scale computationally hard problem, one can solve a large scale problem in online mode.

6 REFERENCES

1. Baruwa O.T., Piera M.A., A coloured Petri net-based hybrid heuristic search approach to simultaneous scheduling of machines and automated guided vehicles. International Journal of Production Research, 2015, 4773-4792.
2. Bocewicz G., Banaszak Z., Multimodal processes scheduling in mesh-like network environment. Archives of Control Sciences, 2015,Vol. 25, No 2, 237-261
3. Bocewicz G., Banaszak Z., Declarative approach to cyclic steady states space refinement: periodic processes scheduling. In: International Journal of Advanced Manufacturing Technology, 2013, Vol. 67, 137–155
4. Bocewicz G., Banaszak Z., Pawlewski P., Cyclic Steady State Space Refinement. Recent Advances In Automation, Robotics and Measuring Techniques, series: Advances in Intelligent Systems and Computing, Springer, 2014, Vol. 267, 11-20
5. Coene S., Arnout A. On a periodic vehicle routing problem, Journal of the Operational Research Society, 2010, Vol. 61(12), 1719–1728
6. Coffman E.G., Elphick M.J., Shoshani A., System Deadlocks (PDF). ACM Computing Surveys. 1971, 3 (2), 67–78
7. Francis P., Smilowitz K., Modeling techniques for periodic vehicle routing problems. Transport. Res. Part B 40 (10), 2006, 872–884
8. Kłosowski G., Gola A., Świć A., Application of Fuzzy Logic in Assigning Workers to Production, Advances in Intelligent Systems and Computing, Vol. 474, 2016. 505-513
9. Ling Y., Chen S., Chiang J. On Optimal Deadlock Detection Scheduling. IEEE Transactions on Computers. 2006, 55 (9), 1178–1187
10. Polak M., Majdzik P., Banaszak Z., Wójcik R.The performance evaluation tool for automated prototyping of concurrent cyclic processes. Fundamenta Informaticae. 2004, 60(1-4), 269-289
11. Relich, M.: Knowledge creation in the context of new product development. In: International Scientific Conference on Knowledge for Market Use, 2015, 834–847
12. Sitek P., Wikarek, J., A Hybrid Programming Framework for Modeling and Solving Constraint Satisfaction and Optimization Problems, Scientific Programming vol. 2016
13. Von Kampmeyer T., Cyclic scheduling problems. Ph.D. Dissertation, Informatik, Universität Osnabrück, 2006

Application of Fuzzy Logic and Genetic Algorithms in Automated Works Transport Organization

Arkadiusz Gola[1,*], Grzegorz Kłosowski[2]

[1] Institute of Technological Systems of Information, Faculty of Mechanical Engineering, Lublin University of Technology, Poland
a.gola@pollub.pl
[2] Department of Enterprise Organization, Faculty of Management, Lublin University of Technology, Poland
g.klosowski@pollub.pl

Abstract. The paper deals with the problem of works transport organization and control by artificial intelligence with respect to path routing for an automated guided vehicle (AGV). The presented approach is based on non-changeable path during travel along a given loop. The ordered set of stations requesting transport service was determined by fuzzy logic, while the sequence of stations in a loop was optimized by genetic algorithms. A solution for both AGV's and semi-autonomous transport vehicles wherein the control system informs the driver about optimal route was presented. The obtained solution was verified by a computer simulation.

Keywords: works transport, control, tandem loop, AGV, path optimization, artificial intelligence, fuzzy logic, genetic algorithms

1 Introduction

Works transport plays a very important role in the logistics of manufacturing systems [1-2]. Effective control of works transport has a beneficial effect on such parameters of a manufacturing system as efficiency, production costs and the size of a production stock [3]. The concept of works transport control is not constrained to fully-automated systems. Bulk and mass production is being replaced by small batch production [4]. Technological progress, globalization, higher consumer demands and competition – all these factors mean that products must undergo constant changes [5]. To meet these demands, manufacturers need to use production systems characterized by a high degree of automation and universality [6-9]. In effect, works transport becomes a vital part of the logistic chain affecting production continuity [10].

Contemporary manufacturing systems are based on the use of various techniques, in which a transport vehicle is sent to a specific pickup and delivery point in order to perform a transport operation [11]. This is usually done by a so-called Automated

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_4

Guided Vehicle (AGV). AVGs are unmanned means of transport which follow routed paths. The interest in applying these vehicles for industrial purposes has been growing steadily for years [12].

The problem of AVG routing and scheduling can be classified as: general routing, route optimization and non-standard vehicle routing comprising [13-14]. Non-standard vehicle routing comprising: methods wherein a route assigned at the beginning of operation remains unchanged until the vehicle completes the route [15-16], time windows methods wherein a route segment can be used by different vehicles in different time windows [17] and methods wherein the use of any route segment is assigned dynamically to a moving vehicle [18-19]. The problems of constant route works transport management pertain to the issues of complex control of AGV systems [20-21] sending AGVs to do transportation tasks – dispatch problems [22-23] and AGV routing and task scheduling [24-25].

This paper presents a predictive system for works transport control which assumes dividing the entire production area (transport) into separate zones operated by single carriers. In effect, the transport system is conflict-free. Route optimization of transport vehicles is done by genetic algorithms. Fuzzy logic was used in the decision process to determine the demand of individual stations for transport services. This solution assumes that every work station requesting transport service is equipped with a separate fuzzy controller describing the station's status.

2 Methodology and experimental details

The study investigates a simulation model of works transport control by an ”intelligent” controller. The investigated objects included a manufacturing system, a transport vehicle and a hybrid system for works transport control based on artificial intelligence techniques. The main objective of the study is to develop a controller for transportation in a specific manufacturing system.

The fuzzy controller module of a single technological machine is a system which receives information in the form of 3-element vector. The input vector $W_x = [x_1; x_2; x_3]$ contains the following elements:

x_1 – progress [%]	$0 \le x_1 \le 100$
x_2 –waiting time for pick up [%]	$0 \le x_2 \le 100$
x_3 – risk	$0 \le x_3 \le 10$

The output is a two-element vector $W_y = [y_1; y_2]$ in which:

y_1 – delivery	$-1 \le y_1 \le 1$	, when	$y_1 > 0$ then Delivery
y_2 – pick-up	$-1 \le y_2 \le 1$	, when	$y_2 > 0$ then Pick-Up

The demand for parts delivery or pick up to or from a work station occurs the moment when the value of output (which is a real number) is higher than zero. However, the higher this value is, the higher the priority of the transport operation requested by a

given work station. The inputs and outputs of the fuzzy controller system are shown in Fig. 1.

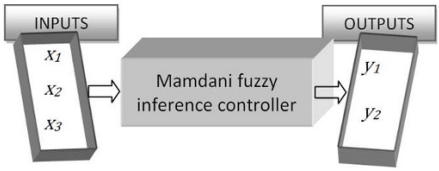


Fig. 1. Inputs and outputs of the fuzzy inference controller

The risk R is a real number contained in the range of $[0, 10]$. The risk depends on both the number of parts left for machining at the moment of delivery and transport batch quantity. Two in the denominator means that is the delivery is at $l_p = p_t$, the risk is equal to 5, so it is mean risk. The rule database applied in the controller consists of six conditional statements which have the following linguistic form:

1. IF (**Progress** is **Low**) OR (**Risk** is **Low**) THEN (**Delivery** is **Redundant**) (significance 1)
2. IF (**Progress** is **Average**) OR (**Risk** is **Average**) THEN (**Delivery** is **Advisable**) (significance 1)
3. IF (**Progress** is **High**) OR (**Risk** is **High**) THEN (**Delivery** is **Urgent**) (significance 1)
4. IF (**Waiting Time for Pickup** is **Short**) OR (**Risk** is **Low**) THEN (**Pickup** is **Redundant**) (significance 0.5)
5. IF (**Waiting Time for Pickup** is **Long**) OR (**Risk** is **High**) THEN (**Pickup** is **Urgent**) (significance 0.5)
6. IF (**Progress is Average**) AND (**Waiting Time for Pickup** is **Long**) AND (**Risk** is **Average**) THEN (**Delivery** is **Advisable**)(**Pickup** is **Advisable**) (significance 0.8)

Figures 2a. and 2b. show the spatial diagrams illustrating relationships between two selected input variables and parts delivery and pickup from the work station. The irregular shape of both planes indicates a complex function which maps inputs as outputs. Therefore, it can be inferred that the task of describing these relationships by a mathematical formula would be very difficult. This fact explains to a high extent the sense and benefits of using fuzzy logic to solve decision problems connected with control processes.

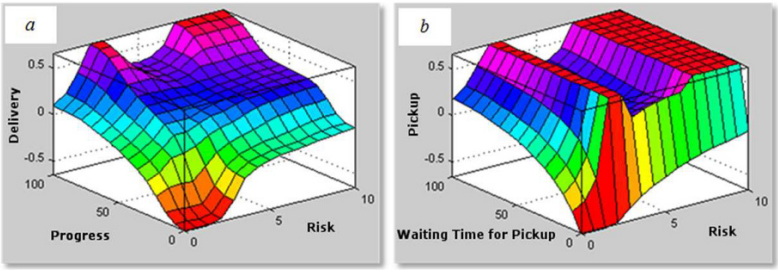


Fig. 2. Effect of progress and risk on delivery (*a*) and the effect of waiting time and risk on pickup (*b*)

One of the key elements of AGV control is selection of an optimal route. The case investigated in this paper has some characteristic features and limitations which distinguish it from a classical travelling salesman problem. First of all, the AGV always departs from one point, i.e. a dispatch station. Secondly, the lowest priority station to be attended as the last one is always imposed on the AVG. What remains is the problem of determining the order of providing service to other stations, that is – a suitable classification of the vector $\boldsymbol{x}_l^1$ (Eq. (2)), containing stations located within the vector $\boldsymbol{x}_l$ (Eq. (1) and (3)) between the dispatch station $n(1)$ and the lowest priority work station $n(i)$.

$$\boldsymbol{x}_l = \{n(1), n(2, n(3), \dots, n(i-1), n(i), n(N_l)\} \tag{1}$$

$$\boldsymbol{x}_l^1 = \{n(2), n(3), \dots, n(i-1), n(i)\}, \boldsymbol{x}_l^1 \in \boldsymbol{x}_l \tag{2}$$

$$\boldsymbol{x}_l = \{n(1), \boldsymbol{x}_l^1, n(N_l)\} \tag{3}$$

The ordering of the assignment vector and stations sequence to the AGV (vector $\mathbf{x}_l$) is done by the GA-based module. The objective function was defined as a cost function [23]. Eq. (5) describes a universal function of cost taking into account fixed costs, operational costs, as well as penalty costs for too early and/or too late arrival at a specific station. The operational costs (Eq. (6)) are proportionate to the time of travel and time spent at a given station.

$$C(\boldsymbol{t}_0, \mathbf{X}) = \sum_{l=1}^{m} c_{f,l} \cdot \boldsymbol{\delta}_l(\mathbf{x}_l) + \sum_{l=1}^{m} C_{t,l}(t_{l,0}, \mathbf{x}_l) + \sum_{l=1}^{m} C_{p,l}(t_{l,0}, \mathbf{x}_l) \tag{4}$$

$$\text{where: } C_{t,l}(\boldsymbol{t}_{l,0}, \mathbf{x}_l) = c_{t,l} \sum_{i=0}^{N_l} \left\{ \bar{T}\left(\bar{t}_{l,n(i)}, n(i), n(i+1)\right) + t_{c,n(i+1)} \right\} \tag{5}$$

Assuming the following:

$$n_0 = 1 \tag{6}$$

$$\prod_{l=1}^{m} \prod_{i=1}^{N_l} \{n(i) - k\} = 0 \quad \forall k = 1,2, \dots, N_l \tag{7}$$

$$\sum_{l=1}^{m} N_l = N \tag{8}$$

$$\sum_{n(i) \in \mathbf{x}_{l,j}}^{m} D\big(n(i)\big) = W_l\big(\mathbf{x}_{l,j}\big) \tag{9}$$

$$W_l\big(\mathbf{x}_{l,j}\big) < W_{c,l} \tag{10}$$

where: $C(t_0, \mathbf{X})$ - total cost [EUR], $\boldsymbol{t}_0$ - vector of departure times of all vehicles from the dispatch station, $\mathbf{X}$ - matrix of assigning all stations to vehicles, $\mathbf{x}_l$ - vector of assignment and stations sequence for the l-th vehicle, $n(i)$ - i-th station visited by a given vehicle, $d(j)$ - the number of dispatch stations, n_0 - the total number of $d(j)$ w $\mathbf{x}_l$, N_l - the total number of all stations attended by the vehicle l, N - the total number of all stations, m - the total number of all vehicles, $c_{f,l}$ - fixed cost of the l-th vehicle (EUR/vehicle), $\boldsymbol{\delta}_l(\mathbf{x}_l)$ - $:= 1$ when the l-th vehicle is used; otherwise, $:= 0$, $C_{t,l}(t_{l,0}, \mathbf{x}_l)$ - operational cost of the l-th vehicle [EUR], $C_{p,l}(t_{l,0}, \mathbf{x}_l)$ - penalty cost for the l-th vehicle [EUR], $c_{t,l}$ - operational cost of the l-th vehicle per minute (EUR/min), $t_{l,n(i)}$ - travel time of the l-th vehicle from the n(i)-th client, $D\big(n(i)\big)$ - demand for pick-up/delivery n(i)-th client [kg], $\mathrm{W}_l(\mathbf{x}_{l,j})$ - load of the l-th vehicle for j-th route (loop) [kg], $\mathrm{W}_{c,l}$ - maximum acceptable payload of the l-th vehicle [kg].

In this paper, the function (4) is a matching function (objective function) optimized by a genetic algorithm. Both the module of fuzzy logic and that of genetic algorithms operate within the simulated manufacturing system. They are part of a works transport control system operating within the manufacturing system. For the purpose of this study, we developed a system enabling fast and effective testing of efficiency of the control system (controller) at different configurations of the manufacturing system.

3 Results and discussion

The experiment verifying effectiveness of the controller was conducted using a specially designed simulation system. One of the simulated system's elements was an intelligent controller developed as part of the study. The experiment was performed on a manufacturing system consisting of 20 technological machines and one transport vehicle. The manufacturing system's parameters (e.g. allocation of technological machines, the number of pickup and delivery points, transport vehicles, organization of parts flow, production parameters of technological machines etc.) were modelled after a real manufacturing system.

Operational parameters of the technological machines used in the experiment are listed in Table 1. The transport batch p_{t_0} is a basic component of a production batch, and it is the initial stock at the start of the simulation. The transport vehicle can deliver parts as a single basic transport batch or its multiplicity.

Table 1. Parameters of the technological machines used in the investigated production system

Machine no.	Basic transport batch p_{t_0} [piece]	Time of machining a transport batch COP		Unit time t_j [s]	Machine no.	Basic transport batch p_{t_0} [piece]	Time of machining a transport batch COP		Unit time t_j [s]
		[s]	[min]				[s]	[min]	
1	50	4000	66,67	80	*11*	20	4000	66,67	200
2	60	4200	70,00	70	*12*	50	5000	83,33	100
3	20	4000	66,67	200	*13*	30	3750	62,50	125
4	20	4800	80,00	100	*14*	10	4000	66,67	400
5	10	4000	66,67	400	*15*	20	4000	66,67	200
6	40	4000	66,67	100	*16*	30	6600	110,00	220
7	40	4000	66,67	100	*17*	10	4000	66,67	400
8	30	3900	65,00	130	*18*	50	4000	66,67	80
9	**20**	**4000**	**66,67**	**200**	*19*	25	3500	58,33	140
10	10	4000	66,67	400	*20*	25	3500	58,33	140

The simulation results were used to verify the operation of the transport system controlled by a fuzzy controller and a genetic route optimizer. The data helped to verify the effectiveness of an intelligent works transport system controller. Below the results for one randomly selected technological machine with respect to its 8-hour working shift (28800 sec.) are presented. This machine is numbered 9 in Table 1. It has average parameters that make part of the simulated manufacturing system.

Fig. 3. illustrates the outputs of ”Delivery” – one of two output signals of the fuzzy controller of the technological machine 9 recorded during the simulation. At the beginning of the simulation the stock of parts for machining was $p_{t_0} = 20$ at the risk R = 5, so the value of the output signal was negative. With decreasing the number of parts for machining (on the input side of the machine), the delivery signal increased. It quickly exceeded the value of 0 and started to acquire positive d values, which signalled the request for starting the operation of another batch of parts for machining.

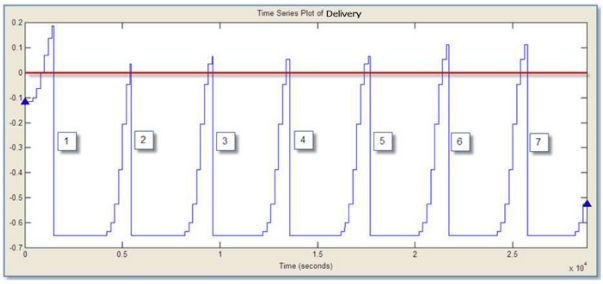


Fig. 3. Outputs of the fuzzy controller for delivery to workstation 9

The extreme case when the vehicle has the task of providing service to all 20 work stations in one path loop can be seen in fig.4. Such case did not occur in the simulation, although theoretically it could have occurred. Fig. 4a. shows the approximate arrangement of work stations in the analyzed manufacturing system. The dispatch station is located at point (0,0). Fig. 4. shows a GA-optimized path loop. Fig. 4c. shows a colour-marked matrix of distance between individual stations.

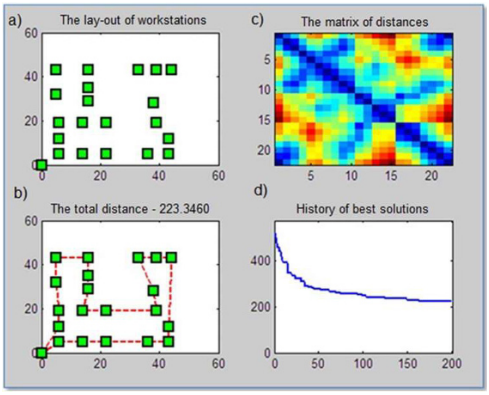


Fig. 4. Optimization of the path loop with all pickup/delivery points

In reality, the path goes through 22 points, as it is the dispatch station that makes the first and the last point, so the distance matrix is a square matrix with the dimensions of 22×22. Fig. 4d. shows the results for individual iterative loops in path optimization by genetic algorithm. The maximum number of iterations was set to 200 in order to prevent excessive prolongation of the computational time and avoid too high reduction

of the algorithm's effectiveness of the algorithm at the same time. This selection was therefore a compromise. However, given the shape of the curve in Fig. 4d, it is possible to believe that the produced solution (the path length equal to 223.346 m) is close to optimal.

4 Conclusions

The numerical results demonstrate that the application of an intelligent controller based on fuzzy logic and an optimizing genetic algorithm is an effective solution for large AGV systems. It was found that the system did not have to optimize a path longer than 13 stations within one loop, despite the fact that it was well capable of providing service to a loop consisting of 20 stations. It was possible to obtain such a low maximum number of stations (13 stations) within one loop in a 20-station system due to the use of an effective method for requesting the transport vehicle to perform pickup and delivery operations. The method for controlling transport service demands per individual technological machines was realized using the Mamdani fuzzy inference system. The system included an innovative element in the form of an additional parameter describing pickup/delivery risk. The experimental results confirm the suitability of both adding the above parameter and of the developed formula for describing this parameter. Computer simulations enable quick determination of the optimal number of parallely serviced zones that the entire manufacturing system must be divided into.

In this paper the hybrid method of Fuzzy Logic and Genetic Algorithms was used. Among the methods of artificial intelligence, the following can be distinguished: particle swarm optimization, coloured Petri nets, Immune Algorithms, Ant Colony Algorithm, Simulated Annealing, Memetic Algorithm etc. The advantage of the proposed method is its versatility, low cost of implementation and the fact that it can be adapted even in already used manned vehicles.

References

1. Burduk, A., Musiał K.: Optimization of chosen transport task by using generic algorithms. Lecture Notes in Computer Science. 9842, 197-205 (2016)
2. Kowalski, A., Marut, T.: Hybrid Methods Aiding Organisational and Technological Production Preparation Using Simulation Models of Nonlinear Production Systems. Lecture Notes in Computer Science. 7209, 259-266 (2012)
3. Gola, A.: Economic Aspects of Manufacturing Systems Design, Actual Probl. Econ. 6 (156), 205-212 (2014)
4. Rudawska, A., Čuboňova, N., Pomarańska, K., Stanečková, D, Gola, A.: Technical and Organizational Improvements of Packaging Production Processes. Advances in Science and Technology. Research Journal. 10 (30), 182-192 (2016)
5. Juszczyński, M., Kowalski, A.: Achieving Desired Cycle Times by Modelling Production Systems. Lecture Notes in Computer Sciences. 8104, 476-486, (2013)
6. Relich, M.: Portfolio selection of new product projects: a product reliability perspective. Eksploatacja i Niezawodnosc-Maintenance and Reliability. 18(4), 613–620 (2016)

7. Relich, M., Świć, A., Gola A.: A Knowledge-Based Approach to Product Concept Screening. Advances in Intelligent Systems and Computing. 373, 341-348 (2015)
8. Kłosowski, G., Gola, A., Świć, A.: Application of Fuzzy Logic Controller for Machine Load Balancing in Discrete Manufacturing System. Lecture Notes in Computer Science. 9375, 256-263 (2015)
9. Jasiulewicz-Kaczmarek, M., Saniuk, A., Nowicki, T.: The maintenance management in the macro-ergonomics context. Advances in Intelligent Systems and Computing. 487, 35-46 (2016)
10. Bocewicz, G.: Robustness of Multimodal Transportation Networks. Eksploatacja i Niezawodnosc – Maintenance and Reliability. 16 (2), 259–269 (2014)
11. Kłosowski, G., Gola A., Świć, A.: Application of Fuzzy Logic in Assigning Workers to Production Tasks. Advances in Intelligent Systems and Computing. 474, 505-513 (2016)
12. Fazlollahtabar, H., Saidi-Mehrabad, M., Balakrishnan, J.: Mathematical optimization for earliness/tardiness minimization in a multiple automated guided vehicle manufacturing system via integrated heuristic algorithms. Robotics and Autonomous Systems, 72, 131-138 (2015)
13. Qiu, L., Hsu, W.-J., Huang, S.-Y, Wang, H.: Scheduling and routing algorithms for AGVs: a survey. International Journal of Production Research. 40 (3), 745-760 (2002)
14. Sitek, P.: A hybrid approach to the two-echelon capacitated vehicle routing problem (2E-CVRP), Advances in Intelligent Systems and Computing. 267, 251–263 (2014)
15. Fan, X., He, Q., Zhang, Y.: Zone Design of Tandem Loop AGVs Path with Hybrid Algorithm. *IFAC-PapersOnLine* ,48 (3), 869–874 (2015)
16. Saidi-Mehrabad, M., Dehnavi-Arani, S., Evazabadian, F., Mahmoodian, V.: An Ant Colony Algorithm (ACA) for solving the new integrated model of job shop scheduling and conflict-free routing of AGVs. Computers and Industrial Engineering. 86, 2–13 (2015)
17. Asef-Vaziri, A., Laporte, G., Sriskandarajah, C.: The block layout shortest loop design problem. *IIE Transactions* 32: 727–734 (2000)
18. Kłosowski, G., Gola, A.: Risk-based estimation of manufacturing order costs with artificial intelligence, 2016 Federated Conference on Computer Science and Information Systems (FedCSIS), 729-732 (2016)
19. Aized, T.: Modelling and performance maximization of an integrated automated guided vehicle system using coloured Petri net and response surface methods. Computers and Industrial Engineering. 57. 822–831 (2009)
20. Nishi, T., Ando, M., Konishi, M.: Experimental studies on a local rescheduling procedure for dynamic routing of autonomous decentralized AGV systems. *Robotics and Computer-Integrated Manufacturing*. 22, 154–165 (2006)
21. Fazlollahtabar, H., Saidi-Mehrabad M., Balakrishnan, J.: Mathematical optimization for earliness/tardiness minimization in a multiple automated guided vehicle manufacturing system via integrated heuristic algorithms. Robotics and Autonomous Systems 72: 131-138 (2015).
22. Rexing, B., Barnhart, C., Kniker, T., Jarrach, A., Krishnamurthy N.: Airline fleet assignment with time windows. Transportation Science. 34 (1), 1–20 (2000)
23. Pillac, V., Gendreau, M., Gueret, C., Medaglia A.L.: A review of dynamic vehicle routing problems. European Journal of Operational Research. 225 (1), 1-11 (2013).
24. Sitek, P., Wikarek, J.: A Hybrid Programming Framework for Modeling and Solving Constraint Satisfaction and Optimization Problems, Scientific Programming vol. 2016, Article ID 5102616 (2016)
25. Taniguchi, E., Shimamoto, H.: Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times. *Transp. Research Part C.* 12, 235–250 (2004).

Quality Assessment of Implementation of Strategy Design Pattern

Rafał Wojszczyk[1]

[1]Departament of Computer Science and Management,
Koszalin University of Technology, Poland
rafal.wojszczyk@tu.koszalin.pl

Abstract. In software engineering, there are many methods and good practices which aim at ensuring quality of developed software. One of these practices is using design patterns. The aim of the article is to show the example of the quality of the implementation of the strategy design pattern and benefits which are given by this pattern. The analysis of the given results can help the software vendor in case of getting rid of the additional costs connected with the development and fixing the defects in software.

Keywords: design patterns, software quality, formal methods, software development

1 Introduction

The commercial software production is always connected with the costs which are paid by the vendor. Particularly when he is producing the business software, which has to be constantly developed to ensure competitiveness towards other products on the market. And when the vendor has to adapt his product to the changing rules. Research and development in the software engineering give many solutions which help to reduce the costs of production, i.e. solutions helping at the beginning of the production process to lower the estimation costs of effort changes in the software [7]. During the whole production process it is popular to use agile factory methods in small teams, which also helps to lower the costs connected with managing the software development. There are also solutions used in the source code of created product, ready components, architecture patterns (e.g. MVC) and design patterns. The last two need own contribution and proper preparation from people who are implementing them.

Work done by people, in this case the source code created by programmers, can include defects. In case of design patterns the defects noticeable by the user can be omitted, whereas defects in design patterns will be noticeable first of all for the vendor. The vendor cannot be sure if despite the defects lower cost of development of the software will be assured. In other words, the vendor is searching for an answer to his question: Whether the implementation quality of the given design pattern will assure him benefits that are expected from this pattern?

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_5

The aim of the article is to present the method of quality assessment of implementation of design patterns, on the example of the Strategy design pattern.

In the second chapter of the article it was briefly explained what is the implementation of design patterns and chosen methods related with searching of the occurrence and verification of the implementation of the patterns were discussed. In the third chapter there were described the strategy pattern and formal model of this pattern. The fourth chapter presents results and their interpretation. The fifth chapter is the summary of the work.

2 Design Patterns and Quality

2.1 Quality of Implementation of Design Patterns

The software vendor, through the implementation of the design patterns, aims at achieving chosen benefits and for majority of the design patterns such are:

- Reusable trusted solutions,
- Way of avoiding chosen defects,
- Lower cost of development and fixing the defects,
- Bigger legibility of the code,
- Substitute for documentation.

The benefits emerging from the implementation of the design patterns contribute to lower the costs of production e.g. new software developer, who knows the design patterns, faster absorb the newly introduced source code (which also contains patterns), than code without patterns. Faster absorbing the code means that new software developer will start to work with maximum efficiency earlier and the minimum efficiency time will be shorter.

The occurrence of the implementation of the design pattern in the source code does not mean that this code ensure mentioned benefits. To be sure, there is a need to check if the quality assessment of the implementation is evaluated by using proper metrics in accordance with the criterion. The criterion of evaluation is a perspective of the expected benefits. To make the evaluation trustworthy it should consider many factors connected with the implementation of the patterns. Basic element in the source code which is evaluated is the structure creating the pattern, e.g. class with the private constructor in the Singleton pattern. Alongside the code which creates the pattern there is a code which is passive in the pattern [4], usually it is processing of the data. The context of the implementation of the pattern is essential but in many times omitted. Elements of the code, where the existence of the pattern is called is the context of the implementation. The metrics in the completed model of evaluation should describe every mentioned elements.

Design patterns are gladly used by software developers both in big and small productive teams. Small teams which work by agile methods often do not dispose of complete documentation of the project of manufactured product, therefore patterns are im-

plemented on the basis of knowledge and experience of the members of the team. However, in many cases, decision about solving the project defect through the implementation of design pattern is made quickly during morning meetings. Mentioned difficulties are determining the problem, so the vendor does not know if what he expects from the implementation of the design patterns will be possible. Therefore from the vendor's perspective, a need for quality assessment of the implementation of design patterns appears, which will help to solve mentioned problem.

2.2 Detection for the occurrence and verification of design patterns

Classic models of quality assessment and metrics are known since many years in the software engineering area. Unfortunately, used for evaluation of the implementation of the design patterns are not sufficient or provide wrong results [4]. Other popular solutions that help in creating software, e.g. Model Driven Architecture, cannot be used by agile teams because of lack of complete project documentation.

Studies directly connected with design patterns very often concern searching of occurrence of patterns in source code only. Information that particular part of the source code contain some amount of patterns is not providing the correctness of implementation, to be precise - correctness of the documentation [6] [5]. Such solution will fail again in case of agile vendor teams. There are undertaken studies, which aim at showing correctness of structural implementation of patterns [1], which also brings too little information and does not include the evaluation.

3 The Model of Quality Assessment of Implementation of Design Patterns

3.1 Model structure

The proposed model of evaluation has been described i.a. in: [8] and [9], where the data models has been explained. In the model it is assumed that the source code is imitated (managed code type .NET or Java, to be precise) into formal model of the program. Templates of the design patterns (i.e. information about implementation variants) were divided into general characteristics and provided with particular metrics, analogically to ISO9126. The norm is described by a tree structure which includes characteristics and in those, recursively, there are subcharacteristics. Characteristics-leaves have metrics - functions assigned to them, which define certain values on the basis of measurable software attributes. The definition of characteristics is informal - it expresses certain intent, while the metrics are formalised.

On the basis of the model it is possible to indicate the quality assessment of implementation of design patterns in the source code and pointing, which elements of the implementation should be corrected to achieve the highest quality in particular evaluation criterion.

3.2 Strategy Design Pattern

Benefits.

Strategy design pattern is one of the popular patterns [2]. The aim of this pattern is [3]: defining algorithm family, encapsulation each one of them and enabling their replacement. This pattern is useful in problems when there is a need to achieve one operation in many different ways. Popular example in literature is encapsulation of different sorting algorithms and their replacement. For the vendor the strategy pattern means following benefits:

- Lower development cost (or bigger accuracy of estimation of the cost) by adding another operations realized with the help of the strategy pattern,
- Repeatability of operations, i.e. operations realized with the help of the strategy pattern will generate the same result regardless of the modulus program in which they will be run in,
- Lower cost of fixing the defects, i.e. the defect should be fixed in one operation, no matter of the cases of use of this operation.

Definition.

The example of the strategy design pattern was evaluated in further part under criterion of evaluation of the lower costs of development. For the vendor it means that the cost of adding new operation to the strategy should stand at $S+n*k$ imaginary work entities. Where S stands for effort of adding new class which implements the strategy.

Effort of the implementation of content of this class, i.e. algorithm realized by new operation is not included in this one. It is assumed that in case $S=1$ (one class with one method). While $n*k$ is the effort needed to make changes in context of pattern implementation, where n is an amount of evoke of the pattern (the declaration of instanton of the strategy).

The template of the Strategy design pattern was imitated in proposed model as a group of characteristics "c": c_1 – interface declaration, c_2 –declaration of operation, c_3 – interface realization, c_4 – implementation of operation, c_5 – context of the implementation of operation, c_6 – initialization and choice of strategy. Mentioned characteristics were detailed by 15 metrics $m_{i,j}$, where i stands for characteristics, and j means another metric. More important metrics in established quality assessment are enclosed in table 1. In established quality assessment the highest result is required. Every lapse from such result indicates additional costs for the vendor.

Table 1 Selected metrics of Strategy design pattern.

Metric	Values (points)
$m_{1,1}$ – access modifier	4 - public, 3 - internal or empty, 2 - protected, 1 - private
$m_{1,2}$ – kind of type	5 – interface, 2 – abstract class, 1 - class
$m_{3,1}$ – kind of type	3 – class, 1 - interface
$m_{1,2}$ – dependencies	5 – if each class realizes interface, 1 - if some class not realizes interface, 0 if any

$m_{4,1}$ – realization of an operation	5 if class realizes (or override) operation, 3 – if class has similar operation, 0 - no operation
$m_{6,1}$ – strategy declaration and initialization	5 – declaration as interface and next initialization of concrete strategy, 4 – declaration and initialization in-line, 1 - no declaration
$m_{6,2}$ – covering of initialized strategy	3 – if all strategies are initialized in one code unit, 1 – if only some

4 Process of quality assessment

4.1 Model of computer software

The most information on programming is included in the source code, which simultaneous shortcoming is a physical representation - these are properly catalogued text files which are more difficult to analyse than a structured formal representation. A formal model of the program provides the automated processing of data concerning the occurring implementation and, additionally, it enables one to remove useless code and unwanted information (e.g. comments written in a natural language, unit tests code).
Basic attributes of object oriented programming were imitated as set: modifiers $MOD = \{static, partial, sealed, final, ...\}$, access modifiers $AMOD = \{public, private, internal, protected\}$, kind of type $KOT = \{class, interface, value, generic, array\}$, kind of instruction $KOI = \{declaration, initialize, called, call\}$. The code of analysed computer program was mapped into set of types T:

$$T = \{t_i \mid i=1, ..., t,\}, \tag{1}$$

where t – number of types, t_i denotes concrete type. Each type t_i:

$$t_i = (FIE_i, PRO_i, MET_i, mod_i, amod_i, name, kot_i), \tag{2}$$

where FIE, PRO, MET are the sets correspondly to fields, properties and methods included in t, $mod_i \in MOD$, $amod_i \in AMOD$, $name \in alphanumerical$, $kot_i \in KOT$. On account of further example, sets FIE and PRO do not require explanation. The set of methods MET_i:

$$MET_i = \{met_{i,j} \mid j=1, ..., met\}, \tag{3}$$

where met – the number of method in t_i, $met_{i,j}$ denotes j-th method in t_i. Each method $m_{i,j}$:

$$m_{i,j} = (MP_{i,j}, rt_{i,j}, ctor_{i,j}, MB_{i,j}, mod_{i,j}, amod_{i,j}, name), \tag{4}$$

where $MP_{i,j}$ – the set of method parameters, $rt_{i,j}$ – returned type, $rt_{i,j} \in T$, $ctor_{i,j}$ denotes constructor (true if method is a constructor, false for otherwise), $MB_{i,j}$ – the set of instructions in method body, $mod_{i,j} \in MOD$, $amod_{i,j} \in AMOD$, $name \in alphanumerical$. Method body is a set of instructions $MB_{i,j}$:

$$MB_{i,j} = \{ins_{i,j,k} \mid k=1, \ldots, ins\}, \quad (5)$$

where *ins* – the number of instructions in *k-th* method in type t_i, $ins_{i,j,k}$ – concrete instruction. Instruction $ins_{i,j,k}$:

$$ins_{i,j,k} = (koi_{i,j,k}, it_{i,j,k}), \quad (6)$$

where $koi_{i,j,k} \in KOI$, $it_{i,j,k}$ instance of type, $it_{i,j,k} \in T$.

4.2 Quality assessment of Strategy pattern

Example program.
Program used in the research is an application project created by students, which is navigating the stepper motors. The Strategy pattern has been used to adapt the application to different actuators, i.e. every student has been using different microcontroller board and developed application should run identically, independently to the equipment. Operation shared by microcontroller boards is: `byte RunCommand(byte cmd, byte args)`, which should be declared in the interface DeviceStrategy. RunCommand operation accepts two byte type parameters and gives back operation result also byte type.

Implementation assessment has been conducted for the next three iterations. In each following iteration students were trying to fix the defects from the previous one. In the example of assessment there has been used metrics and parts of program model, which had the biggest meaning in the particular case.

First iteration.
In the first iteration there were implemented following types: t_1 strategy class, $t_2 - t_6$ strategy implementations (classes responsible for accomplishing operations on different devices) and t_7-t_{12} classes that are using those operations, that is the context of implementation. The remaining types had no relation with implementation of pattern.

First defect was noticed already after using metric $m_{1,2}$ to t_1 = *(Ø, Ø, MET_i, empty, public, DeviceStrategy, class)*, the result of which is 1. Set met_1 includes declaration of operation only. Kind of type t_1 as class instead of interface limits the elasticity of this solution, classes that are implementing strategy (in this case inheriting with t_1) cannot inherit from other classes. Defect emerging from $m_{1,2}$ causes occurring of another defect, which was noticed after using metric $m_{4,1}$ to t_2 = *(Ø, Ø, Ø, empty, public, Atmega8RS, class)*, the result of which is 0, because in t_2 there is no operation. t_2 class (and remaining from t_3 to t_6) inherit from non-abstract class, and it does not need overriding of the methods. This defect will cause call of empty, superior operation instead of the right one.

Another defect is a repeating declaration and initialization of chosen strategy with omitting declaration type t_1, which occurred in a few classes of the implementation context, i.e. $t_7 - t_{14}$. It was noticed after using metric $m_{6,1}$ to body of chosen methods, where another two instructions $ins_{i,j,k}$ = *(declaration, t_2)*, $ins_{i,j,k+1}$ = *(initialization, t_2)*

occurred. The correct implementation is: $ins_{i,j,k}$ = *(declaration,* t_1*)*, $ins_{i,j,k+1}$ = *(initialization,* t_2*)*, it is very important that the superior type t_1 was declared. Such defect will increase effectiveness of adding new operations to existing code because it will be more difficult to find parts which are not directly related to t_1 type.

In summary assessment result for the first iteration was 84 to 97 of possible points in structure creating the pattern and 137 to 234 of possible points in implementation context, this means that development cost of new operations will increase according to *(S+14%) + n*(k+41%)*, where $n = 13$.

Second iteration.

In the second iteration the defects from first iteration were fixed. Class t_1 was replaced by a proper interface, i.e. t_1 = *(Ø, Ø, M, empty, public, iDeviceStrategy, interface)*, and classes implementing this interface obligatory include implementation of expected operation. Call of the strategies were grouped into coherent switch-case block. After the use of metric $m_{6,2}$ to body of chosen methods from classes t_7-t_{14} (code calling strategy) it was noticed that one of the switch-case blocks does not include initialization of all strategies. It was also noticed that there is a declaration of strategy interface (t_1) and initialization of particular class (t_2) so-called inline, however in accordance with: $ins_{i,j,k}$ = *(declaration,* t_1*)*, $ins_{i,j,k+1}$ = *(initialization,* t_2*)*. Omission of remaining classes (t3-t6) affect the lower metric's $m_{6,2}$ result, because it infracts the rule of calling operation from the coherent code part.

In summary assessment result for the first iteration was 97 to 97 of possible points in structure creating the pattern and 129 to 144 of possible points in implementation context, this means that development cost of new operations will increase according to *S + n*(k+10%)*, where $n = 8$.

Third iteration.

In the last iteration the defects from the second iteration were fixed. Examined software got the highest result in all metrics. For the vendor it means that the cost of adding new operation to strategy will be equal to assumed before effectiveness, or $S+n*k$ i.w.e.

5 Summary

In the article chosen problems related to costs of software development were briefly discussed and there were also presented solutions supporting to lower the costs. On the example of strategy pattern there were presented benefits introduced by this pattern.

There was also discussed the author's own model of quality assessment of the implementation of design patterns and described Strategy pattern. Next, there was presented example of assessment that the highest quality of the implementation of patterns will ensure chosen benefits for the vendor.

Further research provide development of the model for information about the quality to give more details and to be comprehensible for the vendor. There is also expected assessment of a few patterns in different iterations of software reference.

6 References

1. Blewitt A., Bundy A., Stark I.: Automatic Verification of Java design patterns, Automated Software Engineering, 2001. (ASE 2001), Proceedings. Conference on 16th Annual International, 2001.
2. Czyczyn-Egird D., Wojszczyk R.: Determining the Popularity of Design Patterns Used by Programmers Based on the Analysis of Questions and Answers on Stackoverow.com Social Network. 23rd Conference Computer Networks, series Communications in Computer and Information Science, Springer International Publishing, Vol. 608, pp. 421-433, Brunów, 2016.
3. Gamma E. et al.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional, Boston 1994.
4. Khaer Md. A. et al.: An Empirical Analysis of Software Systems for Measurement of Design Quality Level Based on Design Patterns. 10th international conference on Computer and Information Technology, pp. 1-6, Dhaka, 2007.
5. Mehlitz P.C., Penix J., Design for Verification Using Design Patterns to Build Reliable Systems, In Proc. Work. on Component-Based Soft. Eng., 2003.
6. Nicholson J. et al.: Automated verification of design patterns: A case study, Science of Computer Programming, Vol. 80, pp 211-222, Elsevier, Amsterdam, 2014.
7. P. Plecka, K. Bzdyra: Algorithm of Selecting Cost Estimation Methods for ERP Software Implementation, in Applied Computer Science, vol. 9, no 2, pp. 5-19, Politechnika Lubelska, Lublin, 2013.
8. Wojszczyk R., Wójcik R., The Model of Quality Assessment of Implementation of Design Patterns, w: Advances in Intelligent Systems and Computing vol. 474, pp. 515-524, Springer International Publishing, Switzerland, 2016.
9. Wojszczyk R., Khadzhynov W.: The Process of Verifying the Implementation of Design Patterns – Used Data Models, in: Advances in Intelligent Systems and Computing, vol. 521, pp 103-116, Springer International Publishing, Switzerland, 2017.

Minimizing energy consumption in a straight robotic assembly line using differential evolution algorithm

Mukund Nilakantan Janardhanan[1*], Peter Nielsen[1], Zixiang Li[2], S.G.Ponnambalam[3]

[1]Department of Mechanical and Manufacturing Engineering, Aalborg University, Denmark
Email: {mnj, peter}@m-tech.aau.dk

[2]Industrial Engineering Department, Wuhan University of Science and Technology, Wuhan 430081, Hubei, China
Email: zixiangli@126.com

[3]School of Engineering, Monash University Malaysia, 46150 Bandar Sunway, Malaysia,
Email:{sgponnambalam}@monash.edu

Abstract. This paper focuses on implementing differential evolution (DE) to optimize the robotic assembly line balancing (RALB) problems with an objective of minimizing energy consumption in a straight robotic assembly line and thereby help to reduce energy costs. Few contributions are reported in literature addressing this problem. Assembly line balancing problems are classified as NP-hard, implying the need of using metaheuristics to solve realistic sized problems. In this paper, a well-known metaheuristic algorithm differential evolution is utilized to solve the problem. The proposed algorithm is tested on benchmark problems and the obtained results are compared with current state. It can be seen that the proposed DE algorithm is able to find a better solution for the considered objective function. Comparison of the computational time along with the cycle time is presented in detail.

Keywords: energy consumption, assembly line layout, cycle time, differential evolution

1. Introduction

Energy is considered to be an important resource in the manufacturing sector and there has been an increasing awareness and need in industry to evaluate methods of reducing the energy consumption due to increasing energy costs as well as corporate responsibility [1]. Assembly lines are considered one of the cost intensive processes in manufacturing industry. It is reported that an efficient production system can be achieved by minimizing the energy consumption [2] and companies are increasingly using robots instead of human labor in assembly lines [3]. The increased use of robots facilitates increased energy consumption and companies are very much required to reduce the energy consumed during their operation. Due to the availability of different types of robots in the market, there is a need for a systematic approach to determine the number as well as the type of robots required to perform a set of assembly tasks [4]. This led to

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_6

the development of a new class of problems named robotic assembly line balancing (RALB) problems. In this area the research mostly addressed the following two problems: RALB-I (Robotic Assembly Line Balancing-Type I), where the objective is to minimize the number of workstations when the cycle time is fixed and RALB-II (Robotic Assembly Line Balancing-Type II) which mainly aims at minimizing the cycle time when the number of workstations are fixed[5]. Recently researchers have reported contributions on RALB with focus on minimizing energy consumption [6] and minimizing assembly line cost [7]. Researchers have also studied different types of layouts mainly: straight and U-shaped. Due to the NP-hard nature of the problem, different optimization techniques such as genetic algorithm, particle swarm optimization and differential evolution have been proposed to solve the problem. The research presented in this paper can be considered as an extension of the work presented in [6]. The major contributions of the work are:

1) To implement differential evolution (DE) a well-known metaheuristic to solve this problem as it is until now not tested in current-state for this problem type. It is also reported in the literatures that DE is very effective to solve NP-hard problems of this nature as shown in [5].
2) The paper presents an algorithm to allocate tasks and robots in a straight robotic assembly line with an objective of minimizing the energy consumption. The algorithm is tested with DE on a set of benchmark problems.
3) The results obtained from the experimental study in this paper are compared with the results obtained using particle swarm optimization algorithm for the same problem as shown in [6].

The proposed model is very much applicable in automobile industries with robot based lines. In conclusion, this paper provides an interesting comparative study on how metaheuristics are effective in solving problem of this nature.

The paper is structured as follows: In Section 2, the problem is defined with the considered assumptions. In Section 3, a detailed procedure of DE is presented and Section 4 discusses the computational study conducted and results are compared with current state. Concluding remarks and future work are presented in Section 5.

2. Problem definition and assumptions

As in Nilakantan, et al. [6], we consider straight assembly lines, where workstations are arranged in a sequential manner and the tasks are allocated to these workstations. The robot with minimum energy consumption is chosen from a set of available robots to be allocated to the workstations. Allocation of tasks is based on the predetermined precedence list of the tasks[8]. Efficiently balancing the workload in each workstation and to select the suitable robot to achieve an efficient balanced assembly line is critical. The objective to be solved in this research is to minimize the total energy consumption of the robotic assembly line. To allow for comparison with existing methods we consider the following assumption as Levitin, et al. [3] and Nilakantan, et al. [6]

- Tasks are not allowed to be split into different stations.

- One workstation can be assigned with one robot. Only one robot can be assigned to a workstation.
- The number of robots and number of workstations are considered to be equal.
- Any robot can execute any tasks assigned.
- Purchase cost is not considered and there is no limitation on the availability of the robots.
- A single model product is assembled.
- Maintenance operations are not considered in this paper.

The solved problem is formulated as in [6] using the same constraints and objective function. This ensures comparability of the metaheuristics.

3. DE for solving RALB problems

In this research, Differential Evolution (DE) algorithm is applied to solve the RALB problem. DE is a recent metaheuristic algorithm proposed by Storn and Price [9] that is being extensively applied to solve real world engineering problems such as scheduling, and design optimization [10]. The algorithm has three main parameters which control the overall search process. DE algorithms are very similar to genetic algorithms; however some differences are present in the crossover and mutation operations. Due to the following advantages, DE has been selected to solve this problem due to following reasons: 1) Fewer parameters to fine tune. 2) Irrespective of the initially set parameters. 3) DE has the capacity to obtain the global best solution. 4) Fast convergence.
The DE procedure starts with a set of initial population named target vectors. In this research these vectors are randomly generated. All the vectors undergo mutation, crossover and selection operations, to generate a new population in such a way to improve the solution quality. A set of donor vectors is generated from the target vectors using the mutation operation. Using the target and donor vector, a crossover operation is conducted to obtain a set of trial vectors. Then a comparison of trial vector and target vector is conducted. The vector with the better fitness value is selected for the next iteration/population. The above mentioned three operations are repeated until a predetermined termination criterion is satisfied. A detailed study is conducted to select the best combination of the parameters used in this research.

3.1 Initialization

One of the main steps in DE is to generate an initial population set [5]. Each member (vectors) of this population is evaluated to obtain a possible solution for the problems. The vector consists of a sequence of numbers (tasks) arranged in such a way that the precedence relations are fulfilled. It is reported in literature that it is good to start the metaheuristic algorithm with vectors that are generated using some heuristic rules rather than randomly. In this research, some heuristic rules reported in literature are applied to generate the initial vectors. By doing so, computation time is reduced. In this research, 25 target vectors are generated as the initial population.

3.2 Mutation Operation

A set of donor vectors is generated using mutation operation on the target vector. Donor vectors are generated by perturbing the population of target vectors. In mutation operation, perturbation is performed by adding the difference between two randomly selected target vectors to a third target vector as shown in Eq. 1.

$$y_{ig} = x_{r1,g} + F(x_{r2,G} - x_{r3,G}), where\ i = 1,...5 \quad (1)$$

F is known as the mutation scale factor, y_{ig} is the donor vector and $x_{r1,\,g}$, $x_{r2,\,G}$, $x_{r3,\,G}$ are the target vectors.

3.3 Crossover Operation

In this paper, crossover operation is done using OX (ordered crossover) operators as proposed by Davis [11]. A set of trial vectors are generated using this operation. To perform crossover operation there is a need for two vectors, in this paper the two vectors chosen are: one from the set of trial vectors and one from the set of donor vectors. Crossover is performed only for a set of vectors in the population based on a crossover rate (C_R).

3.4 Selection Operation

Each target and corresponding trial vector is compared with each other and they compete to get selected for inclusion in the next iteration. The vector with the better function value is copied to the next generation. In this research, selection is based on the lower energy consumption which is the objective function being evaluated.

3.5 Fitness Evaluation- Energy consumption

The objective function which is evaluated in this paper is the energy consumption of a straight robotic assembly line. The workstations are allocated with the tasks to be performed in a balanced way and the robots which perform these tasks with lesser energy consumption are selected. The following steps are involved in the allocation of tasks and robots to a straight robotic assembly line with an objective of minimizing the energy consumption.

Step 1: Based on the task sequence (vector) generated, tasks are allotted to the workstation in straight line layout.

Step 2: Procedure starts by considering an initial energy consumption value E_0.

$E_0 = [\sum_{j=1} \min_{1<i<Nr} e_{hi} / N_w]$, here e_{hi} is the energy consumed by a robot to perform a task,

N_w is the number of workstations and N_r is the number of robots.

Step 3: Based on the E_0 value, tasks based on the tasks sequence gets allocated to the first station. Tasks are added until the sum of energy consumption is less than the considered E_0

Step 4: Next station is opened and the same procedure of allocation of tasks is carried out. This procedure is conducted until all workstations have been utilized.
Step 5: If there are some tasks left to be unassigned within the initial E_0, then E_0 is incremented by 1 and the procedure is repeated until all tasks are assigned.
Step 6: After the tasks are allocated, the procedure of selecting the robots is conducted. The robot which performs the allocated tasks with minimum energy consumption is selected.
Step 7: The sum of the energy consumption of all the workstations is calculated to determine the energy consumption of the assembly line.

Using 11 tasks and 4 robot problems an illustration of the evaluation of the energy consumption of straight robotic assembly line is presented. Table 1 presents the energy consumption and the performance time taken by four robots to perform the 11 tasks. Figure 1 presents the allocation of straight layout configuration with the details of energy consumption in each workstation for a predetermined sequence of tasks based on the precedence relationship *(1-3-2-4-5-6-7-9-8-10-11)*. For this problem initial E_0 is calculated as 35 and it has to be incremented till 43 to get all the tasks allocated to the workstations.

Table 1 Performance time and energy consumption for 11 task 4 robot problem

		Performance Time				Energy Consumption			
Task	**Precedence**	**Robot1**	**Robot2**	**Robot3**	**Robot4**	**Robot1**	**Robot2**	**Robot3**	**Robot4**
1	-	81	37	51	49	20	15	15	17
2	1	109	101	90	42	27	40	27	15
3	1	65	80	38	52	16	32	11	18
4	1	51	41	91	40	13	16	27	14
5	1	92	36	33	25	23	14	10	9
6	2	77	65	83	71	19	26	25	25
7	3, 4, 5	51	51	40	49	13	20	12	17
8	6	50	42	34	44	13	17	10	15
9	7	43	76	41	33	11	30	12	12
10	8	45	46	41	77	11	18	12	27
11	9,10	76	38	83	87	19	15	25	30

In Fig. 1, the robot selected for the workstations are shaded in white and this selection is based on the robot which can perform the allocated tasks with minimum energy. The total energy consumption is calculated to be 150 kJ (26+38+43+43). Based on the performance time data as shown in Table 1, the cycle time of the assembly line is also calculated and determined to be 171. Time at Workstation 1(Robot 3) =51 + 38 = 89. Time at Workstation 2(Robot 4) =42 + 40+25 = 107. Time at Workstation 3(Robot 1) = 77 + 51+43 = 171. Time at Workstation 4(Robot 1) =50 + 45+76 =171.

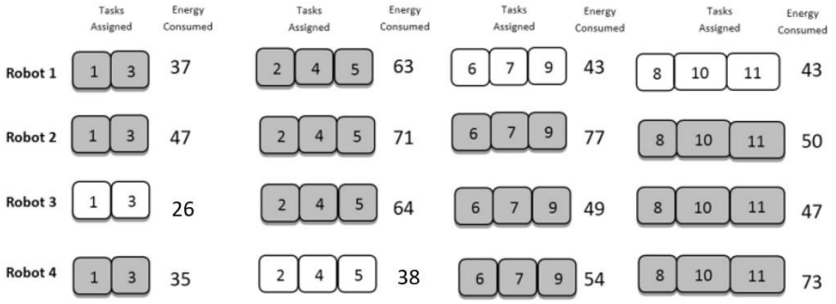


Fig. 1 Allocation of tasks to the workstations with E_0=43

4. Computational Results

To test the performance of the proposed DE algorithm, detailed computational experiments are conducted. Benchmark problems available in literature [6] for RALB problems are utilized to test the algorithm using a formulation matching [6, 7]. Each problem is run ten times and the majority of runs converged to the same solution for all the problems. The best sets of parameters are selected after trial experiments using three dataset problems and are used for all the problems. The parameters used for DE are: number of initial vectors (population size) - 25, number of iterations- 30, mutation factor F=0.5 and crossover rate C_R= 0.7. The results obtained by the proposed algorithm are compared with results reported in [6]. The algorithm used in literature is particle swarm optimization algorithm (PSO). Table 2 presents the comparative results for straight robotic assembly lines using DE and PSO. Table 1 presents the results obtained for small sized datasets (25 tasks problem to 70 tasks problem) as well as presents the results obtained for large sized datasets (89 tasks problem to 297 tasks problem). From the reported results it can be seen that for straight layout proposed DE reports better solutions than PSO for the majority of the problems in both categories. The percentage of energy saving is determined to find the potential of the proposed method. For small sized datasets, an average energy savings of 2.7% is found when using DE model. In case of large sized datasets, an average energy savings of 2.2% could be achieved by using the proposed DE when compared with PSO. For problem 70-7, the solutions obtained using DE and PSO are the same. Solutions (cycle times) obtained by DE is better for 23 datasets and reported same cycle time for 3 datasets out of 32. Cycle times that are the same and higher in the study are presented in bold for clarity. However, since the main objective of the paper is to minimize the energy consumption, no further exploration is required in that direction. The average computational time for the results obtained using DE and PSO are compared and presented in Table 3. The quality of the solution is important when compared to the computational time. It can be seen that for small sized datasets, DE could obtain solutions faster than PSO. However, for large datasets (148 and 297 problems), the computational time of PSO is better than that for DE. This is likely due to the complex selection process involved in DE.

Table 1. Computational Results for proposed RALB problems

Dataset	Proposed DE		PSO [6]		Dataset	Proposed DE		PSO [6]	
	Energy Consumption	Cycle Time	Energy Consumption	Cycle Time		Energy Consumption	Cycle Time	Energy Consumption	Cycle Time
25-3	481	585	494	641	89-8	4922	**612**	5043	562
25-4	340	314	342	314	89-12	5499	**438**	5683	430
25-6	355	207	365	235	89-16	4887	265	5119	340
25-9	235	**143**	248	142	89-21	4182	**232**	4250	206
35-4	1037	518	1072	516	111-9	7131	674	7140	732
35-5	909	369	929	424	111-13	7137	391	7267	396
35-7	974	300	1015	342	111-17	6877	264	6945	280
35-12	658	117	697	160	111-22	6667	215	6909	255
53-5	2662	512	2700	587	148-10	9798	**755**	9840	678
53-7	1970	320	1989	343	148-14	10645	**466**	10654	461
53-10	2148	263	2215	273	148-21	10084	320	10131	335
53-14	2016	202	2177	200	148-29	8334	255	8606	263
70-7	**4146**	**463**	**4146**	**463**	297-19	24351	688	25232	809
70-10	3046	302	3069	290	297-29	24504	435	24970	466
70-14	3802	268	3871	290	297-38	22485	**348**	22862	**348**
70-19	3243	164	3323	255	297-50	21097	**348**	22243	**348**

5. Conclusion

A Robotic Assembly Line Balancing (RALB) problem with an objective of minimizing energy consumption in a straight robotic assembly line is presented. Due to the increasing awareness among industries to be energy efficient, this study will be very useful in automobile industries where robotic assembly lines are extensively used. A differential evolution algorithm is used to solve the proposed problem. The results obtained using DE are compared with results obtained for the same problem solved using particle swarm optimization. Benchmark problems available in literature are used for evaluating the performance. It is found that the proposed DE could find better solutions than PSO for the objective function of minimizing energy consumption of the robotic assembly lines. Cycle times of the allocation done with the objective of minimizing energy consumption are also presented. From the analysis it is concluded that DE is very suitable for solving problem of this nature with some potential challenges in computational time for large problem instances. Parameters for the algorithm are selected through a series of fine tuning experiments and best combination of parameters is reported here. In future, hybrid algorithms along with multi objective robotic assembly line balancing problems could be addressed.

Table 3. Average Computational time of PSO and DE

Datasets	Average Computational Time		Datasets	Average Computational Time	
	PSO	DE		PSO	DE
25	5.3	**4.5**	89	78.5	**70.5**
35	9.2	**7.8**	111	130.3	**127,5**
53	21.3	**18.5**	148	**295.2**	300.5
70	54.3	**50.2**	297	**1512.2**	1530

References

1. Liu, Y., Dong, H., Lohse, N., Petrovic, S., Gindy, N.: An investigation into minimising total energy consumption and total weighted tardiness in job shops. Journal of Cleaner Production 65, 87-96 (2014)

2. Mouzon, G., Yildirim, M.B.: A framework to minimise total energy consumption and total tardiness on a single machine. International Journal of Sustainable Engineering 1, 105-116 (2008)

3. Levitin, G., Rubinovitz, J., Shnits, B.: A genetic algorithm for robotic assembly line balancing. European Journal of Operational Research 168, 811-825 (2006)

4. Relich, M., Pawlewski, P.: A multi-agent system for selecting portfolio of new product development projects. International Conference on Practical Applications of Agents and Multi-Agent Systems, pp. 102-114. Springer (2015)

5. Vincent, L.W.H., Ponnambalam, S.: Scheduling flexible assembly lines using differential evolution. International Conference on Swarm, Evolutionary, and Memetic Computing, pp. 43-50. Springer (2011)

6. Nilakantan, J.M., Huang, G.Q., Ponnambalam, S.: An investigation on minimizing cycle time and total energy consumption in robotic assembly line systems. Journal of Cleaner Production 90, 311-325 (2015)

7. Nilakantan, J.M., Nielsen, I., Ponnambalam, S., Venkataramanaiah, S.: Differential evolution algorithm for solving RALB problem using cost-and time-based models. The International Journal of Advanced Manufacturing Technology 1-22 (2016)

8. Janardhanan, M.N., Nielsen, P., Ponnambalam, S.: Application of Particle Swarm Optimization to Maximize Efficiency of Straight and U-Shaped Robotic Assembly Lines. Distributed Computing and Artificial Intelligence, 13th International Conference, pp. 525-533. Springer (2016)

9. Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization 11, 341-359 (1997)

10. Wang, G.-G., Hossein Gandomi, A., Yang, X.-S., Hossein Alavi, A.: A novel improved accelerated particle swarm optimization algorithm for global numerical optimization. Engineering Computations 31, 1198-1220 (2014)

11. Davis, L.: Applying adaptive algorithms to epistatic domains. IJCAI, vol. 85, pp. 162-164 (1985)

The effectiveness of data mining techniques in the detection of DDoS attacks

Daniel Czyczyn-Egird[1], Rafał Wojszczyk[2]

[1]Department of Computer Engineering,
[2]Departament of Computer Science and Management,
Koszalin University of Technology, Poland
daniel.czyczyn-egird@cicomputer.pl

Abstract. The term of online attacks appeared in public space in the area of computer networks long ago. The effects of these actions can be difficult to rectify and also very expensive. For early detection of such attacks, one can use different methods to analyze the input data generated by the network communication interfaces. The article presented the results of the research on effectiveness of data mining techniques in the detection of DDoS attacks on the selected network resources.

Keywords: computer networks, data mining, DDoS attack, design patterns

1 Introduction

In this day and age of constant development of computers there are many risks related to network attacks aiming at data stealing, destroying and denying access to data [2]. One may avoid many of these attacks by observing the basic principles of information security; however, there are also those attacks with which one has to use special protective strategies and systems which not always guarantee security with one hundred percent certainty.

In the whole list of network attacks, the Distributed Denial-of-Service (DDoS) attack can be found as one of more serious threats and more common attacks aiming at denying access to information services. DDoS attacks consist in generating enormous packages by a great number of systems-agents in order to exhaust computing and communication assets in a quite short time. The most common effect of these actions is making resources and services unavailable to a victim.

The article aims at investigating the relationship and attempting to predict DDoS attack in the selected test area of the computer network. The results of the research may be used to determine parameters and trends having the influence on the behaviour of network traffic, while the scope and methods of the research may show effective usefulness of data mining tools.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_7

2 Characteristics of DDoS attacks

One may observe that today people are being increasingly active in the virtual zone. The Internet has become an integral part of human life. More new social services are being established [5] and they enable users to communicate freely and establish new contacts. Some of the today's activities are slowly becoming essential in an electronic form – it saves a lot of time. Money can be transferred with the use of a computer, and, if help is needed, there is a so-called virtual customer assistant one can use. It all works until there is a banking system failure – an example here can be an unpredicted DDoS attack which cut off the users from their banking systems. Disabled cash machines and payment terminals result in no access to financial funds of many people - panic breaks out.

Therefore, the issues of security in the virtual world are very important. Unawareness of a number of threats possible in the computer sphere may lead to grievous losses, also in the financial dimension. Consequently, one should predict certain dangers in advance and try to minimize losses through quick reactions to undesired actions unless preventing the attacks is possible.

2.1 Specific character of DDoS attacks

Apart from the dangers mentioned in the previous section, network resources are exposed to another type of attacks for which a common thread is denying access to network services. These are Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) attacks [8]. These are the dangers that may result from the attacks: interrupting HTTP requests – problems with access to websites and server applications; interrupting data transferring supported by database servers; stopping print enqueuing in the case of a print server; mail servers cannot send and receive messages; overloading the line of network devices (e.g. router) which may result in shutting local networks from the Internet.

In bygone days, the aim of DoS attacks was disabling the service by means of different mechanisms taking advantage of deficiencies of Transmission Control Protocol/Internet Protocol stack and security vulnerabilities in the specific operating systems. Currently, to deny service, one uses the possibility of generating great traffic interrupting the operation of network applications, server resources and the network itself, and also defects of mechanism of establishing the session of TCP/IP connection. Links or servers are not able to support and process too many requests sent in a short time.

A part of DoS attacks are possible because network hosts use a source IP address or certificates (which can be copied) with authentication. Another problem concerns control mechanisms and routing protocols which use poor authentication methods for the source of information or they are not applied at all. DoS and DDoS attacks may be divided into 3 groups: 1. attacks based on TCP/IP standards – taking advantage of flaws in specification in a particular operating system; 2. attacks based on TCP/IP standards – regardless of an operating system; 3. brute force attacks. Attacks of this kind generate great traffic, seize a network band or also server resources.

2.2 Preventive Measures Concerning on DDoS attacks

The basic defensive technique is filtering of incoming packets – securing the network by using firewall tools including sets of network traffic rules on edge routers, analysing the current flow of packets. The protection consists in blocking the traffic that seems to be suspicious. There are certain good practices applied in the first basic stage of network protection, for instance: disabling the possibility of generating network traffic with source addresses that do not belong to the previously provided pool of IP addresses – the network secured in that way could not participate in the attack; rejection of packets of source addresses that do not belong to our network or to a pool of private class addresses which are not reserved; limitation of attempts of logging to routers; for instance, after three authorisation failures, a particular IP address is temporarily disabled there are no requests from it processed (it is important during DDoS attacks on edge routers); forbidding others to send insets with a broadcast address and rejecting ICMP packets.

Another countermeasure concerning mass requests is an option of configuring services so that the maximum number of simultaneous connections is defined. This setting can be applied to one computer-client (one IP address); however, this will not be effective in case of distributed attack launched by multiple attackers of multiple addresses. An additional problem can be the fact that many hosts can properly use one common address, for example, by using the so-called NAT – network address translation. One should bear in mind that routers' and firewalls' security may fail to provide sufficient protection against more sophisticated attacks.

The countermeasures should be more advanced since firewalls and routers themselves do not protect the user from a compromised client station's access in an internal network. The security should take a form of a vast multi-level architecture that should prevent DDoS attack attempts from the network which includes servers enabling their services. Therefore, another element of defence can be adding subsequent nodes filtering internal traffic which will not allow unverified clients to use the services. Thanks to this action, requests will not be sent to servers directly but will be first filtered and routed to them through indirect stations - agents. When client wants to communicate with server it should first be authorised and determine whether it has proper authorisation for selected services.

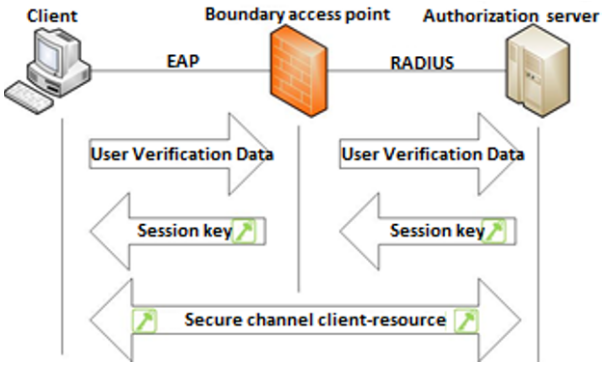


Fig. 1. Typical authorisation protocol architecture

3 Preparations for research

3.1 System of increased resistance

The system of increased resistance to DDoS attacks, unlike typical solutions, e.g. 802.1X standard, which actually does not define an identity verification protocol, has been enriched with a developed algorithm of multi-stage bidirectional authentication [6]. This process takes place in case of every occurring pair of components. The aforementioned requirement that components of each pair identify each other has to be met and will provide the security of sent data even in case of taking over an edge component, that is, agent, by an intruder. This additional assumption makes the authentication procedure more complex. Instead of one-level client identity verification in relation to server, the described protocol offers three times longer process.

The adopted solution involves three-stage procedure of verifying identity. The first stage includes authentication between a client (user component) and an agent (an edge device equivalent) then between an agent and server enabling resource. Only after a positive execution of the two stages, there is a connection established between a client and an authorisation server. To verify identity of each pair of protocol components, the developed authentication algorithm was used. First component (it can be client, agent or server) generates SHA-1 checksum for a sent message. Then the resultant character string is encoded with a private key which is only in the first component. The message and encoded checksum (signature) is sent to the second component which decrypts the signature with the public key of the first component (keys are symmetric, RSA-generated, of 1024 length). Subsequently, the checksum is calculated for the message by means of an identical hash function as in case of first component. If the calculated checksum and the one decrypted from the message are the same, then the first component is authenticated. Otherwise, the request is rejected.

This execution causes that DDoS attack launched by an intruder from the outside may only disable the agent component and, in the worst case scenario, disconnect authorised clients which are connected with it. The main protected resource, that is, the server, stays intact. However, there is still a possibility of attempting DDoS attack with multiple usage of the same certificate from many clients. On the other hand, this threat can be relatively easily eliminated by introducing limitation for the maximum number of simultaneous connections (sessions) for a client.

3.2 Simulation environment

For the simulation purposes, the system of increased resistance to DDoS attacks was implemented as three applications of Microsoft .NET technology [11], in C# language. In the simulation application of this type, there is no typical domain model as in case of business applications, therefore, implementation with the use of architecture patterns as MVC or MVP or a popular three-layer implementation were groundless [12]. However, application modularity was maintained and the selected areas (cryptographic functions, static data, simple data model) were allocated to separate libraries which are shared by all components. What is more, .NET platform standardised libraries were

used, including [11]: System.Net provides essential libraries to handle basic network mechanisms, e.g. operation of a server waiting for connection and a client which will connect to the server; it also provides classes representing an IP address or network data stream object; System.Net.Socket is a subspace for System.Net space. It includes classes responsible for communication by means of sockets; System.Security.Cryptography includes classes responsible for generating keys and handling a digital signature.

Figure 2 shows the first stage of the whole process. The marked piece represents an area of activity when the agent application is waiting in an infinite loop for messages from clients. Each client is handled in a separate thread of application thanks to which the requests are handled concurrently. The parameters recorded for the purposes of further experiment concern one thread of application (that is, one request from a client) or a state of the whole agent application. The parameters selected for the experiment are as follows [9]: DateTime.Now - time of request; difference between GC.GetTotalMemory() readed befor and after handle request; number of active requests; the amount of Process.*memory (NonpagedSystemMemory, PagedMemorySize, PagedSystemMemorySize, VirtualMemorySize, WorkingSet – physical memory usage, PrivateMemorySize), in bytes, allocated for the associated process; process.Threads.Count - gets the number of threads that are running in the associated process; process.*ProcessorTime - gets the total, user or privileged processor time for this process.

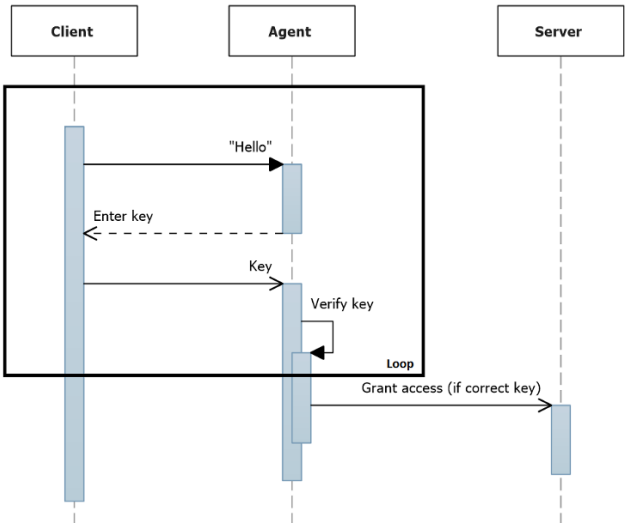


Fig. 2. Sequence diagram with first stage of protocol.

The experiment was done only for the parameters recorded in the first stage, that is, communication between a client and an agent, since here is the place of first contact with a potential intruder. However, on this basis, recording parameters and predicting attacks can be transferred to any other pair of components or even an independent library which can be disseminated as a verified solution.

3.3 Assumptions and the course of the experiment

The assumptions of the experiment aimed at emphasising autonomy of the system, that is, the fact that it can be used as an independent solution, e.g. in dedicated made to measure business applications. The main assumptions are as follows: ignoring network analysers and tools of third entities in general; possibility of using the system from different devices and places, therefore, filtering requests on the basis of an IP address of ACL access list is impossible disabling internal defensive mechanisms which are embedded in the simulation environment; total compromising of a client, that is, taking over all client data by an intruder with a requirement that a real (trusted) client should also have an access.

The course of the experiment consists in performing simulation with the use of the previously described simulation environment. During the 24-hour simulation, one agent component and one server component were assembled; additionally, one of the client components was simulating real traffic while additional instances of the client component simulated DDoS attack according to the aforementioned assumptions.

After the simulation process, a data set was obtained; subsequently, this set was structured for the purposes of Microsoft Excel software which was used to predict the attacks.

4 Research results

4.1 Preliminary data processing

The data set obtained from the simulation included more than 227.000 unique records. Each record has been standardized, because some of the recorded are saved incrementally, i.e. the actual outcome y for a given parameter x is expressed by means of the pattern: $y = x_i - x_{i-1}$.

Thereafter, the data set was divided into three subsets, corresponding to various period of time, which each containing attack. The subsets are characterized by different intensity of the trusted clients' net traffic, that means the requests registered:

1. in the morning i.e. 9:38 – 10:12, with average exertion of the net traffic,
2. in the rush hours i.e. 14:52 – 15:29, with the greatest net traffic,
3. at night i.e. 22:11 – 22:59, with the infinitesimal net traffic of trusted clients.

4.2 The classification of the clients requests

Naive Bayes classifier and K-Nearest Neighbors classification [3] were used for each subset. The analysis was conducted in the public environment of Microsoft Excel using a dedicated additive package XLSTAT [10]. Each classification has been made with the selection of different values for the parameters, according to table 1. Table 2 shows the classification results of the three subsets in regard to two methods of the classification. The present result is the percentage of correctly classified requests.

Table 1. Classification parameters

Naive Bayes classifier		K-Nearest Neighbors	
Ties handling	Rnd. breaker	Number of neighbors	3-10
Prior distribution	Empirical	Metrics/Distance	Euclidean
Smoothing parameter	1	Ties handling	Smallest index
Training set	6336	Training set	12672
Prediction set	1584	Prediction classes	3168
Cross-validation / Number of folds	2	Cross-validation / Number of folds	2
		Weighted vote	Inverse squared distance

Table 2. Classification results

Overview	Naive Bayes classifier	K-Nearest Neighbors	The sample size of data	
			Trusted	Intruders
Set 1	29 %	30,5 %	950	34000
Set 2	33,4 %	36,3 %	17367	76000
Set 3	25 %	25,9 %	352	22000

The analysis of the obtained results indicates that for the test environment of choice and recorded parameters, the most effective is the KNN [1]. However, in the critical case the most effective is the KNN classification with 7 number of neighbors [7].

The effectiveness in the range of 25% - 36,3% is relatively low, compared to other studies [4]. However, it is worth mentioning that, there was an extremely critical case in the used environment test, i.e. the intruder completely intercepted client's identity, not only the verification of IP addresses but also intervals of requests from the client or intruder were omitted. It means that, when all requests are potentially indistinguishable, the proposed method allows to 36,3% of effectiveness.

4.3 The effects of a DDoS attack

During the simulations, each attack at the culminant moment, proved to be effective, i.e. the movement of trusted clients was blocked partially. On the basis of the mechanisms built into the simulation environment, it was estimated that, at the peak of 7% trusted clients did not gain the access to the agent's component , at the same time, up to 38.2% of intruder's requests did not gain the access to the agent. This tendency indicates that, the developed simulation environment may be a fundament to a generative implementation in business. Unfortunately, the sum of abovementioned results indicates that as many as 45.2% of the data has not been registered by the agent's component and is not included in the analyzed data sets. A potential solution of this problem is to launch simulations in the cloud, for example Azure. Then the scalability of the cloud can provide better capability than in the case of a single computer.

5 Summary

In the experiment the simulation environment consisting of three components increasing resistance to DDoS attacks was used. The environment was realized as three independent applications in the Microsoft. NET technology. During the experiment the movement of trusted clients and attacks of intruders on agent's component was simulated. For each registered connection to agent's component selected parameters were saved. Based on registered data the analysis with using techniques of data mining was conducted. The analysis' result shows that KNN method characterizes the most effectiveness in the classification of demands, it will do as device for further research. The correctness of the classification in the range of 25% - 36,3%, taking into consideration, the critical assumptions of the simulations give gratifying results.

The experiment shows that 7% of the trusted clients have been rejected in the culminant moment of the attack. Further work envisage the extension of used data mining techniques and addition more parameters to simulation environment, to ensure greater effectiveness of the attacks detection.

References

1. Ashari A., Paryudi I., Tjoa M.: Performance Comparison between Nave Bayes, Decision Tree and k-Nearest Neighbor in Searching Alternative Design in an Energy Simulation. International Journal of Advanced Computer Science and Applications, Vol. 4, No. 11, pp. 33-39, Bradford UK, 2013.
2. Bandara K.R.W.V., et al.: Preventing DDoS attack using Data mining Algorithms. International Journal of Scientic and Research Publications, Vol. 6, Issue 10, pp. 390-400, 2016.
3. BISHOP C. M., Pattern Recognition and Machine Learning, Springer, 2006.
4. Zhong R., Guangxue Y.: DDoS Detection System Based on Data Mining. Proceedings of the ISNNS 10, pp. 062-065, Jinggangshan, P. R. China, 2010.
5. Czyczyn-Egird D., Wojszczyk R.: Determining the Popularity of Design Patterns Used by Programmers Based on the Analysis of Questions and Answers on Stackoverow.com Social Network. 23rd Conference Computer Networks, series Communications in Computer and Information Science, Springer, Vol. 608, pp. 421-433, Brunow, 2016.
6. Gorski G.: Novel Multistage authorization Protocol. Information Systems Architecture and Technology: Service Oriented Networked Systems. Wroclaw University of Technology. pp. 221-230, Wroclaw, 2011.
7. Hassanat A. B., et al.: Solving the Problem of the K Parameter in the KNN Classifier Using an Ensemble Learning Approach. International Journal of Computer Science and Information Security, Vol. 12, No. 8, pp. 33-39, Ptitsburgh USA, 2014.
8. HeeKyoung Yi, et al.: DDoS Detection Algorithm Using the Bidirectional Session. 18th Conference Computer Networks, series Communications in Computer and Information Science, Vol. 160, pp. 191-203, Ustron, 2011.
9. https://msdn.microsoft.com/en-us/library/ms123401.aspx
10. https://www.xlstat.com/en/
11. Troelsen A.: Pro C# 2008 and the .NET 3.5 Platform. Apress, New York, 2007.
12. Wojszczyk R.: The Process of Verifying the Implementation of Design Patterns Used Data Models. Advances in Intelligent Systems and Computing, Vol. 521, pp. 103-116, 2017.

Part II
Intelligent and Secure Management towards Smart Buildings and Smart Grids

Statistics-Based Approach to Enable Consumer Profile Definition for Demand Response Programs

R. A. S. Fernandes[1], L. O. Deus[1], L. Gomes[2], and Z. Vale[2]

[1] Department of Electrical Engineering, Federal University of Sao Carlos - UFSCar, Sao Carlos, Brazil
ricardo.asf@ufscar.br,
[2] Knowledge Engineering and Decision Support Research Center – GECAD, Institute Polytechnic of Porto – IPP, Porto, Portugal
zav@isep.ipp.pt

Abstract. This paper presents a statistical analysis of a database generated by voltage and current measurements acquired in a laboratorial environment, which simulates a residential kitchen. In this sense, data were acquired during one month in order to verify both the occurrence of errors as well as the possible identification of the loads. Thus, it is intended that the statistical analysis allows the database to be used to the purposes of Demand Response. However, at first, there was an analysis by histograms in order to verify the occurrence of errors on the measurements and then the feature extraction stage. In the sequence, these features were used to define decision rules that could perform the identification of loads. The results obtained demonstrated an average precision rate of more than 90%.

Keywords: statiscal analysis, non-intrusive load monitoring, confidence intervals, demand response

1 Introduction

The methodologies aimed at the identification of loads, also known as Non-Intrusive Load Monitoring (NILM) have gained visibility in recent years [1]-[4], mainly due to the programs of Demand Response. One of the concerns in that area is the idealization of methods to optimize the relationship between energy demand and their hourly prices [5]-[6]. In this way, it is important to define an approach that is efficient, where many studies have been proposed for NILM, among which one may highlight [1], [2] and [4].

Hence, in [1], the authors propose a non-invasive method that can be used both for monitoring purposes and for the identification of residential appliances. The meter used requires a sampling rate that ensures the acquisition of at least 10 samples per second. It is worth mentioning that this method divides the appliances into three major categories that represent sets of appliances. However, it is noted that some of the appliances were represented by categories, probably because the appliance is on the edge between the two categories. Thus, the

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_8

authors used a Linear Discriminant that was possible to obtain a precision rate above 80

At the work proposed by [2], the authors precisely show that data from smart meters can be used in order to disaggregate the load consumptions and then improve the decisions to be taken by consumers to reduce their energy costs. The proposed methodology was based on the verification of events in data windows where the following tasks are performed: edge detection, trend signature, time signature, and sequence signature. The decision is made based on the responses obtained for each window. Using this method, the authors obtained a precision rate above 90

In a recent method proposed by [4], the authors employ a temporal multi-label technique to identify the loads of residences in France, where the data was sampling in 10, 30 and 60-minutes interval. Thus, the authors evaluate the identification of individual loads and combinations of two loads at the same time.

Following the above context, this paper proposes an analysis of a laboratorial database through statistical tools, with two objectives. The first objective is the histogram analysis in order to verify the occurrence of possible errors on the measurements. The second objective is the proper identification of loads through decision rules.

In the sequence, Section 2 presents the composition of the laboratorial database and the infrastructure that was used to acquire the data; the histogram analysis are performed in Section 3; the statistical methods used to generate the decision rules are presented in Section 4; the results as well as the discussions are presented in Section 5. Finally, Section 6 is intended to present the conclusions of this paper.

2 Laboratorial Database Composition

It is important to mention that the laboratorial database was implemented in SQL Server, where data could be obtained by remote queries. However, such a database could only be created by means of a laboratorial infrastructure which is based on three-phase power analyzers from the Electrex (Femto D4 model). After obtaining the current and voltage signals, these are evaluated in order to calculate some variables (extracted features), such as: RMS (Root Mean Square) voltage and current; power factor; apparent, reactive and real power; total harmonic distortion of voltage and current; and neutral current. Importantly, all these variables can be obtained for each of the three phases and a time stamp is assigned for each measurement. In addition, each set of measurements is available in 10-second intervals.

The generated variables are available via Modbus (RS-485) communication channel. Thus, a Programmable Logic Controller (PLC) receives the data and sends them through an Ethernet communication channel to the Database Server. In order to clarify this explanation, a block diagram representing this system is shown in Figure 1.

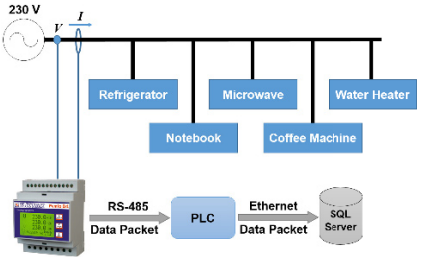


Fig. 1: Block diagram representing the laboratorial infrastructure.

3 Statistical Analysis Using Histograms

This first analysis by means of histograms was based on the possibility of the occurrence of errors or inconsistencies in some data due to inaccurate measurements. Such analysis were performed for all measurements, which were gathered for individual and combined loads, i.e., when a load is turned on together with other load(s). In the sequence, such analysis can be verified for each individual load.

3.1 Microwave

After the calculation of mean of RMS current for the microwave, it is possible to define it as 5.7986. Thus, Figure 2(a) shows the histogram of the data analyzed. It is worth mentioning that the mean have the above values, mainly due to the sample that is highlighted in the histogram. However, it is believed that this sample has been obtained in a situation that caused a reading error by the analyzer. Another analysis was the combination of the Microwave with the other loads present in the laboratorial environment, which can be viewed on the histogram of Figure 2(b).

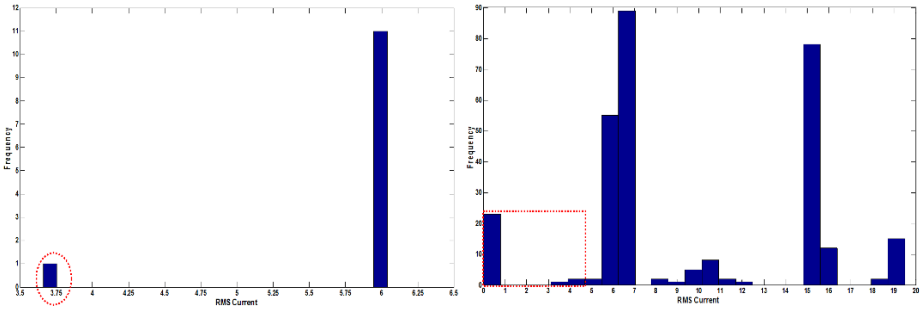


Fig. 2: Analysis through histogram: (a) only the microwave; (b) microwave combined with other loads.

In these cases, it was noted that there were probably also errors of measurement for the highlighted samples, as these are below the average for the RMS

current of the cases where the microwave was individually observed. It is important to mention that the same analysis were performed for all other obtained variables.

3.2 Coffee Machine

Since the microwave presented such errors, the same tests were replicated for the coffee machine. Thus, Figure 3(a) shows the histogram analysis for it. Again, some measurements were found to be erroneous. Figure 3(b) presents the histogram that allows the detection of erroneous measurements for the combination of the coffee machine with the other loads.

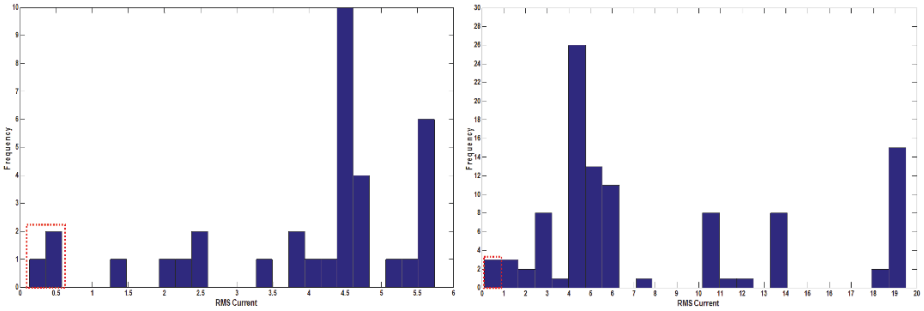


Fig. 3: Analysis through histogram: (a) only the coffee machine; (b) coffee machine combined with other loads.

3.3 Water Boiler

It is important to mention that the water boiler is the highest power load and, consequently, it is the load with the highest individual current. Another factor that should be commented is that the water boiler is a resistive load, differing from the other loads under analysis. Due to its higher current, it was possible to observe that its mean is 9.3292. Thus, it is noted that measurement errors are virtually eliminated, that is, very probably due to its high current when compared with the other loads. As expected, when performing the histogram analysis for the water boiler when individually fed Figure 4(a) or fed in combination with other loads Figure 4(b), the errors were null.

3.4 Refrigerator

Finally, the refrigerator is a very complex case to have its measurement errors assessed, as its RMS current is low when compared to other loads and, for this reason, its mean is 0.42. Thus, by means of Figure 5(a), it was noted that some samples had a rather inconsistent value for when the refrigerator was individually fed, because its RMS current went from a mean of 0.42 A to almost 5 A. Thus, it is concluded that such a measurement would only be possible due to the following

causes: error by the CT transducer, a possible transient event occurred in the electrical network or a transient event caused by the refrigerator compressor.

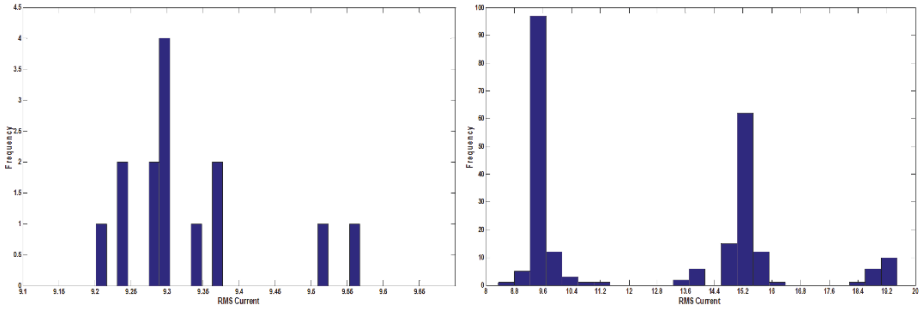


Fig. 4: Analysis through histogram: (a) only the water boiler; (b) water boiler combined with other loads.

As for Figure 5(b), it shows the analysis performed for the refrigerator combined with other loads, where there is a difficulty in establishing possible erroneous measurements, since the other loads have high currents when compared to the refrigerator. Therefore, in these cases, no erroneous sample could be detected.

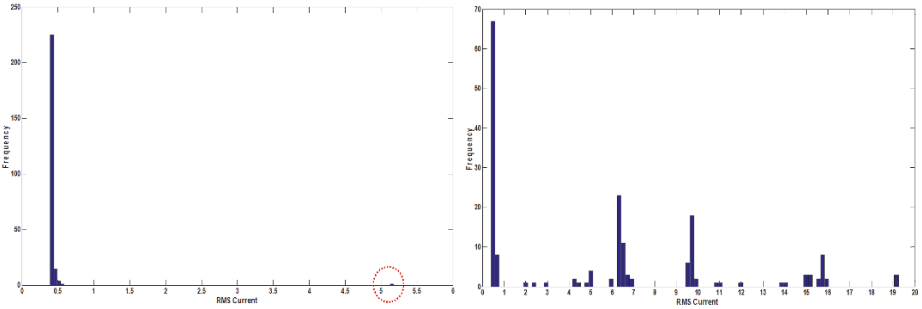


Fig. 5: Analysis through histogram: (a) only the refrigerator; (b) refrigerator combined with other loads.

After the histogram analysis, it was possible to detect samples that could generate an incorrectly identification by means of the decision rules. Thus, measurements that presented errors were excluded from the database so that the statistical calculations were redone considering, for such purposes, only reliable measurements.

4 Decision Rules Based on Statistical Analysis

Based on the analysis performed through histograms, some errors in the measurements became evident, which were then eliminated from the database to

generate decision rules that might characterize the loads (individually and combined). Thus, in principle, it was analyzed the performance for the following statistical calculations:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i, \tag{1}$$

where μ represents the mean calculated for a set of data x that is limited between the interval [1, N];

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}, \tag{2}$$

where σ represents the standard deviation calculated for a set of data x;

$$y = max(x), \tag{3}$$

where y represents the maximum value obtained for a set of data x;

$$z = min(x), \tag{4}$$

where z represents the minimum value obtained for a set of data x;

$$P = \mu \pm \frac{\sigma(y_{\alpha/2})}{\sqrt{N}}, \tag{5}$$

where P is the confidence interval for a set of data with N samples and $y_\alpha/2$ can be found from tables that is used to define the confidence level. So, in this paper, it was used a confidence level of 90%.

In this sense, decision intervals were generated for individual and combined loads. However, it is important to mention that three methods were firstly analyzed:

- based on mean - its respective standard deviation were used to define the upper and lower limits;
- based on confidence intervals - their upper and lower limits were defined by the sum of the confidence interval and the standard deviation;
- based on maximum and minimum values - it was used just the maximum and minimum values of the data set.

After an extensive analysis of the behaviour of these three statistical analysis, it was observed that a hybrid approach would be more suitable to identify the loads where such method should combine the best characteristics of them. Thus, it appears that the combination of decision rules generated by maximum and minimum values and confidence intervals could ensure better identification for most loads and their combinations.

It is noteworthy that, during the implementation of the decision rules, it was noted that the most important feature to identify the loads were the apparent power. Following, the reactive power has become a very important feature for the refinement of the identification of water boiler and its combination with other loads. Besides these two variables, total harmonic distortion of current, RMS current and power factor were also used.

5 Performance Analysis

This section presents and discusses the results of each of the proposed statistical methods, but focused on the hybrid method. Thus, the precision rates for the identification of each load – Refrigerator (RE), Microwave (MI), Water Boiler (WH) and Coffee Machine (CM) – and their combinations can be seen in Table 1.

Table 1: Precision rates obtained for each load and their combinations.

Loads	**Precision Rate [%]**			
	Mean	**Conf. Intervals**	**Max. and Min. Values**	**Hybrid**
without load	100	100	100	100
RE	85.9	88.1	99.6	100
MI	35.5	8.3	20.7	93.4
WH	67.3	67.3	100	100
CM	66.1	79.8	100	80.7
RE+MI	77.8	77.8	88.9	100
RE+WH	55.9	79.4	100	100
RE+CM	0	0	0	0
MI+WH	59.3	59.3	100	100
MI+CM	75.0	0	85.0	85.0
WH+CM	45.0	50.0	95.0	100
RE+MI+CM	31.6	31.6	57.9	63.2
RE+WH+CM	82.4	64.7	94.1	94.1
RE+MI+WH	81.8	81.8	90.9	100
MI+WH+CM	0	0	0	25.0
RE+MI+WH+CM	81.8	81.8	100	100

The identification with precision rate below 80% by the hybrid method for the coffee machine occurred due to the fact that, in some cases, their measurements are superposed to the microwave measurements. This superposition is characterized by current peaks (wide range of amplitudes) in certain times of its operation. However, by contrast the Maximum and Minimum Values method with the Hybrid method, note that the first one has a better identification response for the coffee machine. Therefore, this better identification results in an incorrectly identification of the microwave by the Maximum and Minimum Values method (20.66%), especially, when compared to the response of the hybrid method (93.39%). Thus, to ensure a better overall response of the hybrid method, it was necessary to make such a concession, even by the fact that the coffee machine can be regarded as a less relevant load on the Demand Response context.

The most complex identification to be performed is the combination of the refrigerator and the coffee machine, because the amplitude variations of current peaks that the coffee machine presents superpose the refrigerator measurements

generating a misclassification by all methods. Therefore, the incorrectly identification can be a potential problem because the refrigerator is more relevant than the coffee machine and, for this reason, new features must be obtained and evaluated to improve the identification.

Almost methods present an unknown response for the identification of refrigerator, microwave and coffee machine when combined. This way, it is possible to create new and specialized rules considering other variables to be analyzed. Finally, the precision rate about of 25% for the combination between microwave, water boiler and coffee machine, all methods present an erroneous response due to consider a combination where the four analyzed loads are turned on.

6 Conclusions

This paper presented a methodology sufficiently effective, since it is based solely on statistical calculations to set decision rules. Such rules could be embedded in hardware, because the proposed approach does not depend on high processing power and also does not require large memory. Therefore, it is believed that the presented method is suitable for applications that may use the load identification to the purposes of Demand Response.

7 Acknowledgements

This paper was supported by FAPESP (grant number 2016/00641-4), CAPES and CNPq.

References

1. Z. Wang and G Zheng. Residential Appliances Identification and Monitoring by a Nonintrusive Method. *IEEE Transactions on Smart Grid*, 3(1):80–92, 2012.
2. M. Dong, P. C. M. Meira, W. Xu, and C. Y. Chung. Non-Intrusive Signature Extraction for Major Residential Loads. *IEEE Transactions on Smart Grid*, 4(3):1421–1430, 2013.
3. R. A. S. Fernandes, I. N. Silva, and M. Oleskovicz. Load Profile Identification Interface for Consumer Online Monitoring Purposes in Smart Grids. *IEEE Transactions on Industrial Informatics*, 9(3):1507–1517, 2013.
4. K. Basu, V. Debusschere, S. Bacha, U. Maulik, and S. Bondyopadhyay. Non Intrusive Load Monitoring: A Temporal Multi-Label Classification Approach. *IEEE Transactions on Industrial Informatics*, 11(1):262–270, 2015.
5. A. Anvari-Moghaddam, H. Monsef, and A. Rahimi-Kian. Optimal Smart Home Energy Management Considering Energy Saving and Comfortable Lifestyle. *IEEE Transactions on Smart Grid*, 6(1):324–332, 2015.
6. C. Vivekananthan, Y. Mishra, G. Ledwich, and F. Li. Demand Response for Residential Appliances via Customer Reward Scheme. *IEEE Transactions on Smart Grid*, 5(2):809–820, 2014.
7. M. Dong, P. C. M. Meira, W. Xu, and W. Freitas. An Event Window Based Load Monitoring Technique for Smart Meters. *IEEE Transactions on Smart Grid*, 3(2):787–796, 2012.

Feature Extraction-Based Method for Voltage Sag Source Location in the Context of Smart Grids

F. A. S. Borges[1], I. N. Silva[1], and R. A. S. Fernandes[2]

[1] Department of Electrical and Computer Engineering, University of Sao Paulo - USP, Sao Carlos, Brazil
fabbioanderson@gmail.com,
[2] Department of Electrical Engineering, Federal University of Sao Carlos - UFSCar, Sao Carlos, Brazil
ricardo.asf@ufscar.br

Abstract. The location of the voltage sag sources corresponds to an important task for the Power Quality area. However, this is not a trivial task due to the need of many monitoring devices. Therefore, with the data management from smart meters installed in distribution feeders, decision support tools that solve this problem become viable. Thus, this paper proposes an algorithm that determines the area where the voltage sag source is located. For this purpose, it was necessary to extract features from smart meters' voltage signals. In the sequence, we analyze the relevance of each feature to establish the most significant of them. In this way, the smart meter could extract these features and send them to the utility. At the utility side, the proposed algorithm will estimate the region where the voltage sag source is located. The location procedure is performed by cross checking the most relevant features and the network topology.

Keywords: voltage sag, power quality, distribution feeders, disturbance location

1 Introduction

In accordance with standards and/or recommendations under the Power Quality (PQ) area, a voltage sag is characterized by the magnitude decrease of the nominal voltage, where the values are situated between 0.9 and 0.1 pu, with durations of 0.5 cycles to 1 minute. Moreover, such disturbance corresponds to a great part of PQ events in primary distribution feeders. However, this kind of disturbance brings significant economic losses, mainly, for industrial consumers [1]. For this reason, it is extremely important for utilities that voltage sags are identified and located. This way, some actions should be taken to mitigate the disturbance.

In this sense, some methods have been published and present good results to identify PQ disturbances [2]-[3]. On the other hand, methods designed to locate PQ disturbances are not common due to its complexity [4]-[5]. The process of

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_9

location for voltage sags is not a trivial task, because this disturbance occurs in a short time and propagates throughout the feeder. Moreover, the deployment of smart meters, with capabilities of performing PQ analysis and provide data through communication channels, demands a high cost [4].

Thus, this paper presents a method that determines the location of voltage sags by analyzing the features of the disturbance that propagates in the network and is found in various parts of the system. Thus, the events detected near the origin present similarities, whereas there are dissimilarities in the disturbances detected in a position distant to the source. The features studied in this work could be easily provided and implemented by a monitoring system based on smart meters.

This paper is organized in seven sections. Section 2 presents related works. In Section 3, it was presented the IEEE 13-buses, which was simulated by using the ATP software. In the sequence, Section 4 demonstrated the aspects of feature extraction stage. Section 5 is dedicated to present the proposed method. Finally, the results and conclusions are respectively presented in Sections 6 and 7.

2 Related Works

The methods related to the location of PQ disturbances are intended to indicate the position in the feeder where the disturbance source is connected. Considering the techniques found in the literature [5]-[6], it is observed that there is a difficulty in determining the location of the source of voltage sag. So, these methods typically use information acquired from one monitoring point to determine if the source is located downstream or upstream of the meter. Therefore, for such methods have more accurate results, it is necessary to obtain information from other monitoring points.

In [7], the method estimates the voltage at each node using a multivariate regression model. The node that has the maximum voltage deviation and minimum standard deviation is determined as voltage sag source location. This method was validated by using a system with 9 nodes (6 meters), and another system with 30 nodes (8 meters). The Discrete Wavelet Transform was used in [8] to preprocessing signals generated after the switching of capacitor banks. The energy of each wavelet leaf was adopted as inputs to a hybrid system composed of Artificial Neural Network and Principal Component Analysis.

The method proposed in [9] is based on a current variation index calculated for each meter. According to the authors, the nodes (with meters) that present a voltage sag are near to the source of the event. Following this premise, the proposed index was calculated before and after the PQ event. Thus, voltage sag source is found by comparing the current deviation rates.

It is noted that the above methods fail to achieve satisfactory coverage and development, as they require detailed system information but also rely on signal processing calculations that must be provided by the monitoring system, which increases the implementation cost.

3 Simulation of Voltage Sags by Using ATP

In order to evaluate the proposed algorithm, it was necessary to model and simulate the IEEE 13-Bus Test Feeder [10] through the ATP (Alternative Transients Program) software. Having modeled this system, simulations were performed by changing the system's behavior to generate voltage sags with different durations and magnitudes. Thus, it was possible to obtain a database to validate the proposed algorithm.

It is worth mentioning that all voltage sags were generated as a result of single-phase short-circuits, involving the phase C (with different fault impedances). In this sense, the voltage sag occurs until the elimination of the short-circuit.

4 Analysis of Features Extracted

A study about the event properties, as evidenced in each system branch, was conducted to better define heuristics about the system behavior in the presence of voltage sags. The knowledge obtained in this phase is used to implement the location method of the causes of sags.

4.1 Feature Extraction Foundations

A feature extractor should reduce the input pattern vector (i.e. the waveform of the detected disturbance) to a smaller dimension, preserving all useful information of the original vector. The extracted features should highlight the similarities between the disturbances detected around the location region, and present dissimilarities for the measured disturbances that are distant from the origin. In addition, the feature extraction will minimize the computational effort of the method. The features analyzed were RMS (V_{RMS}), Peak Value (V_{peak}), Shannon Entropy (S) and Form Factor (FF), respectively expressed by equations 1 to 4:

$$V_{RMS} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} v_i^2}, \tag{1}$$

$$V_{peak} = max(v_1, v_2, .., v_i, .., v_N), \tag{2}$$

$$S = -\sum_{i=1}^{N} v_i^2 log(v_i^2), \tag{3}$$

$$FF = \frac{\frac{1}{N}\sum_{i=1}^{N} v_i}{V_{RMS}}, \tag{4}$$

where v_i represents the i^{th} sample of instantaneous voltage and N is the quantity of samples for the segment.

4.2 Disturbances Propagation Analysis

After defining the metrics to be calculated, some fault scenarios were generated and the features were extracted from segmented data (voltage sag disturbance detected in each fault simulation). Three simulations are subsequently presented to illustrate how the disturbance propagation analysis was performed according to the fault location. It is noteworthy that node 652 has been neglected since it only has phase A and only single-phase faults in phase C were used in the tests.

Table 1 presents data of the features calculated in each node to a simulated fault applied at node 611, located at the end of the system. According to this table, it is noted that the lowest values of V_{RMS}, V_{peak} and FF were obtained at node 611 and node 684 that are close to the source. On the other hand, the S feature does not show this similarity as the lowest value was found in node 650, which is located far from the point where the fault occurred.

The next simulation analyzed corresponds to a fault situation in node 671. The data of the calculated features are also presented in Table 1. In this second scenario, it is noted that the calculated features presented very close values when compared among nodes upstream of the faulty node. Moreover, downstream nodes have distinct values calculated at the faulty node, thus emphasizing an area of similarity. The FF feature was the one that best represented this similarity.

Table 1: Features extracted from smart meters' signals for a fault applied in the nodes 611 (columns 2 to 5) and 671 (colunms 6 to 9), considering the phase C.

Nodes	V_{RMS}	V_{peak}	S	FF	V_{RMS}	V_{peak}	S	FF
611	0.467	0.652	65.903	-0.0660	0.4190	0.5766	67.71	-0.0728
632	0.554	0.776	67.897	-0.0239	0.4864	0.6774	68.38	-0.0451
633	0.553	0.775	67.926	-0.0239	0.4859	0.6767	68.40	-0.0450
634	0.553	0.775	67.928	-0.0239	0.4858	0.6766	68.40	-0.0450
645	0.553	0.775	67.914	-0.0238	0.4862	0.6772	68.39	-0.0449
646	0.553	0.774	67.937	-0.0238	0.4858	0.6766	68.41	-0.0448
671	0.503	0.702	67.790	-0.0366	0.4187	0.5776	67.79	-0.0728
675	0.503	0.701	67.797	-0.0366	0.4186	0.5776	67.79	-0.0726
680	0.503	0.702	67.790	-0.0366	0.4187	0.5776	67.79	-0.0728
684	0.485	0.677	67.014	-0.0508	0.4188	0.5771	67.75	-0.0728
692	0.503	0.702	67.790	-0.0366	0.4187	0.5776	67.79	-0.0728
650	0.626	0.881	63.086	-0.0046	0.5961	0.8364	63.70	-0.0080

Table 2 presents the feature values calculated for a fault applied at node 632. At this point, the disturbance has very close amplitude features, except for the voltage signal measured at the substation where the sag measured was less severe when compared to the others. Node 611 presented the lowest values of V_{RMS} and V_{peak}. On the other hand, the lowest value of FF was evidenced at nodes 671, 692 and 680, which are located closer to node 632, which originated the disturbance. For this scenario, it is concluded that the disturbance flows more strongly from the fault location to the ends, and the downstream nodes present a less severe disturbance.

This analysis enables the understanding of the system behavior in the occurrence of a voltage sag disturbance originated in different parts of the system. The obtained heuristics are summarized below:

- The features V_{RMS}, V_{peak} and FF can define the degree of similarity between the disturbances, namely, nodes near the disturbance origin have similar feature values;
- The disturbance is "perceived" with greater intensity in the nodes located upstream of the location of the source that caused the event;
- When the cause of the disturbance is located in a node close to the substation, the disturbance is "perceived" with similar features in all system nodes;
- When the cause of the disturbance is located in the system ends, the disturbance is "perceived" with greater intensity at the place of origin of the disturbance.

Table 2: Features extracted from smart meters' signals for a fault applied in the node 632 (phase C).

Nodes	V_{RMS}	V_{peak}	S	FF
611	0.520	0.741	68.642	-0.0460
632	0.528	0.754	68.626	-0.0456
633	0.528	0.753	68.636	-0.0456
634	0.528	0.753	68.637	-0.0456
645	0.528	0.753	68.633	-0.0454
646	0.528	0.752	68.641	-0.0453
671	0.521	0.743	68.674	-0.0460
675	0.521	0.743	68.682	-0.0459
680	0.521	0.743	68.674	-0.0460
684	0.521	0.742	68.660	-0.0460
692	0.521	0.743	68.674	-0.0460
650	0.614	0.867	64.447	-0.0085

It is important to highlight that the most relevant feature is the FF, because for a purely sinusoidal signal, it assumes a value of approximately zero. However, for waveforms containing oscillatory transients, the FF presents abrupt changes in its value. Therefore, by using the FF, it is possible to determine the oscillatory transient at the beginning of the signal and highlight the magnitude variation when the voltage sag occurs.

5 Proposed Method

Based on the analysis previously presented, the proposed method uses the FF as the information provided by smart meters. Thus, it has been aimed at defining a region in which the voltage sag source is located. Accordingly, the first step is to configure the topological structure of the feeder as a tree data structure. Thus, each node of the tree will represent a feeder node, as shown in Figure 1.

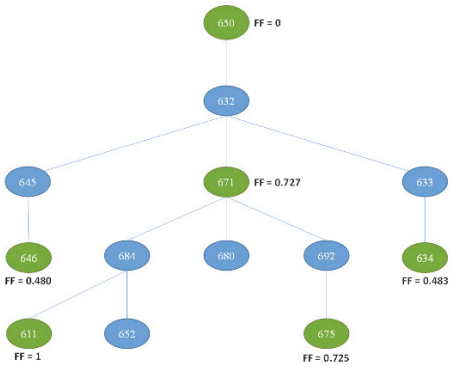


Fig. 1: IEEE 13-bus test feeder represented as a tree data structure with smart meters allocated.

The second step defines the position of each smart meter, which is based on the power quality engineer's knowledge. In the sequence, the values of the FF of each smart meter are received and normalized. Thus, it is possible to know how the disturbance sensitizes the meters. This normalization procedure is calculated based on equation 5:

$$Norm_i = \frac{FF_{max} - FF_i}{FF_{max} - FF_{min}}, \tag{5}$$

where i represents the smart meter's ID. Highlighting that the form factors are calculated by the meters only when the voltage sag is detected, i.e., considering the segment. Consequently, there is a need to calculate an adaptive threshold (Equation 6) that corresponds to the summation of FF of each smart meter divided by the number of meters (N_{meters}):

$$Threshold = \frac{\sum_{i=ID} FF_i}{N_{meters}}. \tag{6}$$

In this way, the smart meters that present FF higher than the threshold are considered within the region where is located the voltage sag source. After obtaining the list of smart meters, it is possible to eliminate the other meters and to generate a newer tree. Thus, the root node will be defined by the smart meter with the highest FF (equal to 1). Finally, the region where is located the voltage sag source is obtained by calculating the difference between the FF of the root node and the FFs of its leaf nodes.

6 Results and Discussions

In order to conduct a well-defined study relating to the proposed method, it was simulated some cases of single-phase short-circuits. The meters are placed at the ends because of possible consumers, as it is usually where there is the greatest concentration of loads that would be affected by the disturbance. The substation meter, on the other hand, is commonly found (node 650). Finally, a last meter was allocated at node 671 due to this being a point of many system derivations. Such allocations may be seen in Figure 1 (nodes highlighted in green).

Table 3 shows the results, where the first column shows the node where is applied the fault. The second and third columns of this table correspond to the calculated threshold and the region determined by the algorithm, respectively.

Table 3: Voltage sag sources location.

Faulty Node	Threshold	Location
611	0.416	611-684-671
632	0.827	671-632-645-646
633	0.582	634-633-632-645-646
634	0.547	634-633-632-645-646
645	0.452	645
646	0.357	646
671	0.689	671-692-675-684-611
675	0.547	671-692-675-684-611
680	0.452	671-680
684	0.541	611-684-671-692-675

The first case corresponds to a short-circuit at the node 684 and, for this case, the adaptive threshold calculated is 0.541. Following the previously determined steps, the meters that have values greater than the threshold were installed on the nodes 611, 675, and 671. Thus, the new tree data structure can be generated, where the node 611 is the root (highest FF) and the nodes 675 and 671 are the leaves, as shown in Figure 2a. From this new data structure, the path having the greatest difference, in terms of FF, determines the location area. Therefore, it is noted that the method infers a voltage sag source between the node 611 and the node 675. However, among them are the nodes 684 (faulty node), 671, and 692. Figures 2b and 2c present the location areas for voltage sag sources when the short-circuit is applied at nodes 633 and 632, respectively.

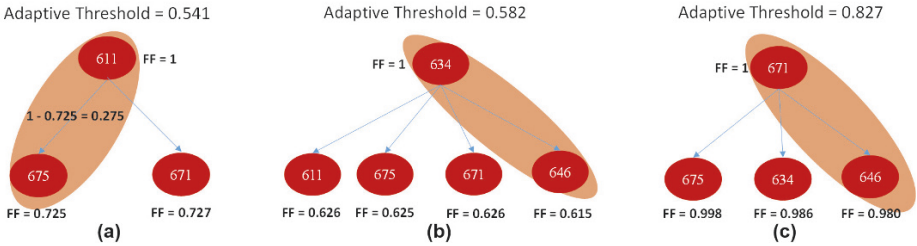


Fig. 2: Tree data structure representing the results: (a) fault at node 684; (b) fault at node 633; and (c) fault at node 632.

It is observed in Table 3 and in Figure 2 that the algorithm could correctly determine the location of the voltage sag sources. However, for some cases, this area involved some nodes of the analyzed feeder. Nevertheless, in these cases the accuracy of the proposed algorithm is still good, since it will the distance between nodes are small, finding a reduced region.

7 Conclusions

This paper presented an algorithm that represents an alternative to determine the location of voltage sag sources. For this purpose, smart meters connected to the primary feeder during short-circuit events calculate some features. Among these features, its was verified the relevance of the Form Factor. Based on the results, it can be concluded that the method is effective in detecting voltage sag source location. In this way, it is clear that the proposed algorithm could be applied to the Smart Grid context. For future work, it would be important to idealize a method based in machine learning for at estimate a threshold using the value RMS. After set this threshold the method will be testing in others distribution feeders.

8 Acknowledgements

This paper was supported by FAPESP (grant number 2013/16778-0), CAPES and CNPq.

References

1. R. C. Dugan, M. F. McGranaghan, and H. W. Beaty. Electrical Power Systems Quality. 3rd Edition. New York, 2002.
2. M. S. Manikandan, S. R. Samantaray, and I. Kamwa. Detection and Classification of Power Quality Disturbances Using Sparse Signal Decomposition on Hybrid Dictionaries. *IEEE Trans. on Instrumentation and Measurement*, 64(1):27–38, 2015.
3. O. P. Mahela, A. G. Shaik, and N. Gupta. A Critical Review of Detection and Classification of Power Quality Events. *Renewable and Sustainable Energy Reviews*, 41:495–505, 2015.
4. G. W. Chang, J. P. Chao, H. M. Huang, C. I. Chen, and S. Y. Chu. On Tracking the Source Location of Voltage Sags and Utility Shunt Capacitor Switching Transients. *IEEE Trans. on Power Delivery*, 23(4):2124–2131, 2008.
5. N. Hamzah, A. Mohamed, and A. Hussain. A New Approach to Locate the Voltage Sag Source Using Real Current Component. *Electric Power Systems Research*, 72(2):113–123, 2004.
6. M. Erol-Kantarci, and H. T. Mouftah. Energy-Efficient Information and Communication Infrastructures in the Smart Grid: A Survey on Interactions and Open Issues. *IEEE Communications Surveys & Tutorials*, 17(1):179–197, 2015.
7. A. Kazemi, A. Mohamed, H. Shareef, and H. Raihi. Accurate Voltage Sag-source Location Technique for Power Systems Using GACp and Multivariable Regression Methods. *Int. Journal of Electrical Power & Energy Systems*, 56:97–109, 2014.
8. Y. Y. Hong, and B. Y. Chen. Locating Switched Capacitor Using Wavelet Transform and Hybrid Principal Component Analysis Network. *IEEE Trans. on Power Delivery*, 22(2):1145–1152, 2007.
9. G. W. Chang, J. P. Chao, S. Y. Chu, and C. Y. Chen. A New Procedure for Tracking the Source Location. *IEEE PES General Meeting*, 1–4, 2007.
10. IEEE Distribution Planning Working Group Report Radial Distribution Test Feeders. *IEEE Trans. on Power Systems*, 6(3):975–985, 1991.

A multi-agent system for energy trading between prosumers

Meritxell Vinyals[1], Maxime Velay[1], and Mario Sisinni[2]

[1] CEA, LIST, 91191 Gif-sur-Yvette, France
meritxell.vinyals@cea.fr, maxime.velay@cea.fr
[2] R2M Solution, F-91405 Pavia, Italy
mario.sisinni@r2msolution.com

Abstract. Local energy communities are identified as a promising approach to efficiently integrate distributed generation whereas keeping costs down for prosumers. In this context, we propose a multi-agent system to collectively optimise the energy flows of a local community of prosumers. The novelty and strength of our approach resides in the use of decentralised decision making algorithms, based on the alternating direction method of multipliers, to orchestrate the demand and supply of a large number of homes. Our preliminary results show how the proposed approach can significantly increase the self-consumption level of the community while significantly reducing the energy bills of its members.

Keywords: multi-agent systems, distributed optimisation, local energy exchange, alternating direction method of multipliers, prosumer community.

1 Introduction

Smaller commercial and residential customers account for between one-third and one-half of the total electricity consumption in many markets and represent the greatest need and opportunity for demand response [3]. Furthermore, more and more consumers are becoming *prosumers* (i.e. producing their own energies with renewable green technologies such as solar energy) and the trend is expected to continue with all new construction being able to produce energy on-site.[3] However, since distribution networks are traditionally not designed to cope with large amounts of generation, increasing distributed generation (DG) is expected to create congestion and voltage problems if not properly handled. In this context, *local energy communities* (LECs) are identified as a promising approach to integrate DG whereas maintaining energy security and keeping costs down for consumers [7]. In such communities, prosumers can locally trade their excess of energy, reducing in this way the transmission losses which occurs over long distances and making the system as a whole more efficient. Indeed, we can already find across EU some real examples of successful LECs, as for example the EWS Schönau[4] in Germany, or the *REScoop 20-20-20* project[5] in Belgium.

[3] The EU Parliament's Committee on Industry, Research, and Energy declared that all buildings built after 2018 will be able to produce their own energy on-site.
[4] https://www.ews-schoenau.de/ews/international/
[5] https://rescoop.eu/

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_10

Nevertheless several research challenges remain to be addressed before the widespread of these local energy communities can take place. First, it is crucial for the success of such communities to provide prosumers with *individual incentives* to reward their participation. For example, recent studies stated that 51% of people interviewed would be motivated to get involved in community energy if they could save money in their energy bill [4]. However, more traditional approaches for trading locally produced energy have focused on optimizing the overall system, generating solutions that may result in weak financial incentives of some prosumers [1, 9]. Second, prosumers will only be comfortable sharing their data if they are confident that these are stored securely and exchange in a way that safeguards their *privacy*. Last but not least, the distribution level of next generation electric grids will include a large number of active devices, whose control and scheduling significantly increases the complexity of the corresponding energy management problem compared to traditional grids [2, 6]. Consequently, traditional approaches that commonly solve this problem in a centralised fashion will become computationally impractical due to its lack of *scalability*.

In this paper we address these challenges by proposing a Multi-Agent System (MAS) to collectively optimise the energy flows of a local community of prosumers. The decentralised decision making is based on the Alternative Direction Method of Multipliers (ADMM) algorithm. As we show in Section 3, the ADMM protocol enables a multi-period local market, where the preferences of the prosumers and their cost structures are reflected in their private utility functions and the interactions between them take place through price signals. Concretely, this paper makes the following contributions:

1. We model the local energy exchange problem by means of an energy cooperative network.
2. We distribute this cooperative network among different agents and we use the ADMM algorithm as a coordination mechanism among these agents.
3. We empirically evaluate our approach via simulations. Our results show how the proposed approach can significantly increase the self-consumption of the community of prosumers whereas appreciably reducing their energy bill.

This paper is structured as follows. We first give, in Section 2, some background on energy coordination networks and the ADMM algorithm. Afterwards (Section 3) we present the MAS local energy exchange model. Finally, we validate the proposed approach in simulation (Section 4), to later conclude in Section 5.

2 Background

2.1 Energy coordination networks

A cooperation network [8] is composed of a set of actors, A, a set of nets, N, and a set of terminals, T. A terminal is a connection point, a connection between one actor and one net. This is depicted in Fig. 1 where nets are represented by grey dashed rectangles, terminals are represented by lines and actors by circles. In an *energy coordination network*, a net is an agent that represents a virtual

energy exchange zone (i.e. where energy exchange among actors is negotiated and prices determined) whereas an actor is an agent that owns one or multiple transfer points, each modelled as a terminal.

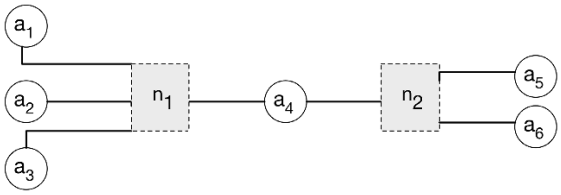


Fig. 1: Example of a coordination network.

In an energy coordination network, each terminal $t \in T$ has associated a power schedule $(p_t(1), \ldots, p_t(H)) \in \mathbb{R}^H$ over a time horizon $H \in \mathbb{N}^+$. Then, for each $\tau \in [1, H]$, $p_t(\tau)$ is the power consumed (if $p_t(\tau) > 0$, otherwise produced) by actor a through terminal t, during the time slot corresponding to τ.

For each actor $a \in A$, we use a to refer to both the actor itself as well as to the set of terminals associated with it, i.e., we say $t \in a$ if terminal t is associated with actor a. The set of all power schedules associated with actor a is denoted by $p_a = \{p_t | t \in a\}$, which we can associate with a $|a| \times H$ matrix. In addition to $|a|$ terminals, actor a is also associated with an objective function $f_a : \mathbb{R}^{|a| \times H} \to \mathbb{R}$, where $f_a(p_a)$ is the exploitation cost of actor a for the power schedule p_a. Moreover, every actor has a set of constraints, denoted as C_a, that p_a should satisfy in order to be a feasible planning.

Analogously to actors, every net $n \in N$ has $|n|$ terminals, an objective function $f_n : \mathbb{R}^{|n| \times H} \to \mathbb{R}$ and a set of constraints C_n that $p_n = \{p_t | t \in n\}$ should satisfy in order to be a feasible planning. Since each net $n \in N$ models an energy exchange zone, the set of constraints C_n should always include the energy balancing condition, i.e. $\sum_{t \in n} p_t(\tau) = 0, \ \forall \tau \in [1, H]$.

2.2 Alternating direction method of multipliers

ADMM [2] is an algorithm that allows solving the underlying optimisation problem of a coordination network in a distributed way. Formally, an energy coordination network models the following optimisation problem:

$$\begin{gathered} \min_{p \in \mathbb{R}^{|T| \times H}} \sum_{a \in A} f_a(p_a) + \sum_{n \in N} f_n(p_n) \\ \text{subject to } \forall a \in A: \ p_a \in C_a, \forall n \in N: \ p_n \in C_n \end{gathered} \tag{1}$$

To form the augmented Lagrangian, the nets objective functions are defined over a duplicated copy of the original variables (i.e. denoted as $\dot{p}$) and the equality constraint $(p = \dot{p})$ is relaxed via a Lagrange multiplier. Then, the ADMM algorithm consists on applying the following three steps at each iteration $k + 1$:

The *actor-minimization* step (i.e. parallelized among actors):

$$\forall a \in A, \quad p_a^{k+1} = \arg \min_{p_a \in C_a} f_a(p_a) + \frac{\rho}{2} \sum_{t \in a} ||p_t - \dot{p}_t^k + u_t^k||_2^2 \tag{2}$$

The *net-minimization* step (i.e. parallelized among nets):

$$\forall n \in N, \quad \dot{p}_n^{k+1} = \arg\min_{\dot{p}_n} f_n(\dot{p}_n) + \frac{\rho}{2} \sum_{t \in n} ||p_t^{k+1} - \dot{p}_t + u_t^k||_2^2 \tag{3}$$

The (price) *scaled dual variables* update (i.e. parallelized among nets):

$$\forall n \in N, \quad u_n^{k+1} = u_n^k + (p_n^{k+1} - \dot{p}_n^{k+1}) \tag{4}$$

3 Local energy exchange community model

In this section, we formulate the energy exchange problem in a prosumer community as a coordination network. Fig. 2 illustrates this coordination network with an example of four prosumers that join the same community. The figure also depicts how the coordination network is distributed among the two types of socio-economical agents that participate in the optimisation, namely: (i) the HEMS (Home Energy Management System) agent, that carries and out an in-home optimisation behind the meter with the objective of minimising the prosumer energy bill and (ii) the AGR (Aggregator) agent, which optimises the energy sharing among prosumers by enabling an energy exchange zone (i.e. the AGRNet) at the community level. The decentralised local energy exchange results from applying the ADMM negotiation protocol over this network.

Next section details the in-home coordination network that lies in the boundaries of the HEMS agent whereas Section 3.2 does the same for the first level of the hierarchy, the AGR agent.

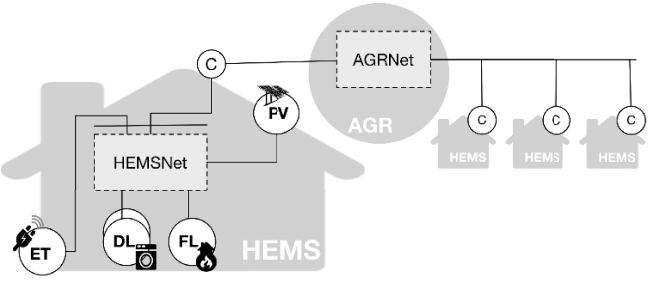


Fig. 2: Different HEMS coordination networks connected through an AGR.

3.1 The HEMS agent: the in-home coordination network

The prosumer, represented by the corresponding HEMS agent, represents the smaller unit of the community and also the individual provider of flexibility. Table 1 lists the different type of in-home devices considered in our validation. As shown in Fig.2, each in-home device is modeled as an actor, encoding its local costs and constraints. Notice also that the HEMS coordination network does not consider solutions in which the prosumer imports energy from the grid (through

Table 1: Types of in-home devices considered in validation.

PV	A solar photovoltaic generator that is expected to produce some energy but whose production cannot be controlled.
FL	A fixed (inelastic) load that requires a fixed consumption but that does not provide any flexibility and cannot be controlled.
DL	A deferrable (i.e. shiftable) load that requires its execution to be done by a certain time specified by the user.
ET	A connection to an external source of power used to model the connection of the prosumer with the energy supplier (and the corresponding individual supplier tariff applied).

the ET) and exports at the same time the energy produced from the PV. Since the connection with the supplier is represented by a single terminal (i.e. between ET and HEMSNet), the power planned through this connection at a given time slot can be positive or negative but never both.

3.2 The AGR agent: the community coordination network

The AGR agent is represented by a net through which it creates a local energy exchange zone for community exchanges. From an optimisation perspective, if the prosumer joins a local community it adds also a connection with the AGR. By means of this AGR terminal the prosumer can: (i) import energy for consumption of the in-home device(s); and (ii) export energy from its local PV production.

This AGR connection should remain different from the connection that the prosumer has with the supplier (the ET connection). In other words, the model needs to guarantee that prosumers will not sell to other actors (i.e. other prosumers in the community) the energy imported from the grid (i.e. imported by means of their corresponding supplier tariff). To avoid those scenarios, we need to change the type of HEMSNet to not only consider the typical balancing constraints but also to check that no transfer of power takes place between the AGRNet and the ET (i.e. to check that for any time slot, both power schedule have the same sign). This new type of net is represented in the figure as a dashed rectangle with double line on the terminals that should have the same sign.

4 Simulation results

In this section we validate our approach in order to demonstrate that the MAS proposed is an effective optimisation tool for local energy management and local balancing. Firstly, we explain the details of our experimental setup in Section 4.1. Secondly, we analyze our results in Section 4.2.

4.1 Setup

Agents and message exchange have been implemented with the JADE platform[6]. Each scenario is composed of one local community (i.e. a set of prosumers that

[6] jade.tilab.com/

joined the same community through the same aggregator). Notice that to generate benefits from the community optimisation two conditions are needed, namely that some prosumers in the community: (a) have local generation (e.g. PV); and (b) have deferrable loads. Otherwise, no energy exchange can be established among prosumers and hence the optimised profiles will be the same as those obtained when optimising each home individually. Therefore, the scenarios will focus on configurations in which prosumers have local (PV) generation and deferrable loads. In particular, each problem instance models a local community of eight prosumers. Each prosumer is assigned a predicted fixed consumption profile and, six of them, a PV generation profile. For predicted fixed consumption profiles, we use the two Elexon domestic[7] load profiles [5], chosen to represent large populations of similar customers in UK. For the PV generation profiles we used real-data provided in the context of the Mas2tering FP7 project[8] coming from a low voltage grid in Cardiff (UK). Table 2 specifies for each prosumer which Elexon domestic profile was assigned (i.e. the profile is specified by the prosumer class, the season and the day type) and, in the case of PV production, which PV profile uses (i.e. specified by the season and the number).

Table 2: Characterisation of prosumers used in experiments.

	Elexon profile			
Prosumer	Class	Season	Day type	PV
A	ECONOMY7	WINTER	WEEKDAY	WINTER-0
B	ECONOMY7	WINTER	SATURDAY	WINTER-1
C	ECONOMY7	AUTUMN	WEEKDAY	AUTUMN-0
D	ECONOMY7	AUTUMN	SATURDAY	AUTUMN-1
E	UNRESTRICTED	WINTER	WEEKDAY	WINTER-2
F	UNRESTRICTED	WINTER	SATURDAY	-
G	UNRESTRICTED	AUTUMN	WEEKDAY	AUTUMN-2
H	UNRESTRICTED	AUTUMN	SATURDAY	-

Moreover, each prosumer has a (ET) connection that models a two-band TOU tariff[9] (with 0,158 €/kWh and 0,11 €/kWh as prices for peak and off-peak respectively) with export prices[10] (0.05€/kWh exported) in the case of PV. Then, for each instance of this initial scenario, we generate different configurations varying the percentage of the fixed load that can be shifted (i.e. by adding deferrable loads). In more detail, while the total flexible load is still not totally distributed, we create a new deferrable load by randomly sampling the minimum and maximum time slot (i.e. in which the load can be scheduled) and by randomly generating a power profile for the load (considering that the total of this power profile should be less than the remaining flexible amount to be distributed). The time horizon was chosen to be $H = 96$, corresponding to 15

[7] Among the eight Elexon profile classes only two refer to domestic (residential) consumers, namely class 1 (unrestricted) and class 2 (economy7).

[8] www.mas2tering.eu

[9] This is a real tariff currently offered by EDF in France, `https://particuliers.edf.com/fichiers/fckeditor/Particuliers/Offres/CGV_CRE/CRE_TB.pdf`

[10] The export tariff follows the prices in UK feed-in tariff for exportation: https://www.gov.uk/feed-in-tariffs/overview

minutes intervals for 24-hour schedule, with the time period $\tau = 1$ corresponding to midnight. For each configuration, we generated and solved 25 problem instances to compute the average reduction and the standard error of the mean ($\sigma_M = \frac{\sigma}{\sqrt{N}}$) as a measure of the variance (error bars).

4.2 Results

We analyse the results obtained regarding the reduction in the energy bill and the self-consumption for different proportions of deferrable loads.

Reduction in the energy bill after community optimisation Fig. 3a depicts the relative reduction in the daily community energy bill for different percentages of deferrable load. The reduction is computed w.r.t the baseline case in which prosumers only optimise their in-home coordination network without being connected to an aggregator (i.e. the sum of the payments of the prosumers without joining a community). Observe that the higher the percentage of load that can be shifted (i.e. the flexibility available in the aggregator's community), the higher the savings obtained by prosumers that joined the community.

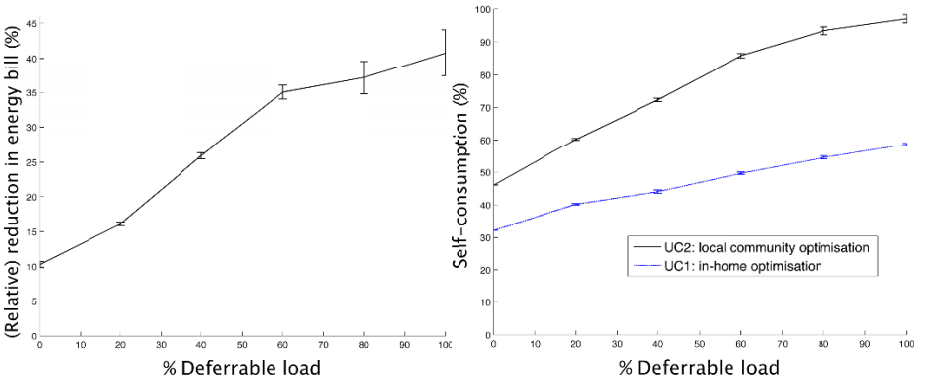


Fig. 3: a) Relative reduction in the community energy bill and b) community self-consumption as a function of the percentage of the load that can be shifted.

Self-consumption after community optimisation Fig. 3b shows the percentage of self-consumption of the aggregators portfolio for the case of only in-home optimisation and for the case of community optimisation. We observe that the percentage of self-consumption at community level is significantly larger after local community optimisation as a result of community exchanges. Moreover, we also observe that the larger the percentage of deferrable load the larger the increment of self-consumption. For the 10-20% shiftable load scenario, the average self-consumption at community level after local community optimisation is around 60% whereas only around 40% if only applying in-home optimisation.

5 Conclusions

This paper formulated a MAS that allows prosumers in a community to locally trade their excess of energy and to schedule their shiftable loads when more local energy is available. In particular, we formulated the problem as a coordination network distributed among two types of agents, the HEMS and the AGR agents, each with its own goals and acting to pursue its own objectives. Following ADMM exchanges, each HEMS agent enrols into a negotiation process with the AGR agent, exchanging energy plans and price signals needed to reach an agreement and hence, respecting the information privacy of participants. The proposed framework was evaluated in simulation scenarios composed of a community with eight prosumers. Simulation results show how prosumers can get extra savings of around 15% when 20% of their load is shiftable while still improving self-consumption (20% increase) as a result of their exchanges in the local community.

As a future work, we plan to explore the ability of the local community to not only trading locally produced renewable energy but also to provide ancillary services (voltage regulation, congestion management,. . .) to local grid operators.

Acknowledgments

Authors would like to acknowledge the support of the European Union under the FP7 Grant Agreement no. 619682 (MAS2TERING project).

References

1. Borenstein, S., Holland, S.: On the efficiency of competitive eletricity markets with time-invariant retail prices. RAND Journal of Economics 36(3), 469–493 (2005)
2. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. Foundations and Trends in Machine Learning 3(1), 1–122 (2011)
3. Cooke, D.: Empowering customer choice in electricity markets. IEA Energy Papers 2011/13, International Energy Agency (October 2011)
4. Department of Energy & Climate Change: Community energy strategy. Policy paper URN 14D/019, UK Government (January 2014)
5. Elexon: What are the profile classes?, `www.elexon.co.uk/knowledgebase/profile-classes/`
6. Eto, J.H., Thomas, R.J.: Computational needs for the next generation electric grid proceedings. Tech. Rep. LBNL-5105E (April 2011)
7. Faruqui, A., Sergici, S.: Household response to dynamic pricing of electricity: a survey of 15 experiments. Journal of Regulatory Economics 38(2), 193–225 (2010)
8. Kraning, M., Chu, E., Lavaei, J., Boyd, S.P.: Dynamic network energy management via proximal message passing. Foundations and Trends in Optimization 1(2), 73–126 (2014)
9. Mihaylov, M., Jurado, S., Moffaert, K.V., Avellana, N., Nowé, A.: NRG-X-Change - A Novel Mechanism for Trading of Renewable Energy in Smart Grids. In: SMARTGREENS 2014. pp. 101–106 (2014)

Smart Grids Data Management: A Case for Cassandra

Gil Pinheiro, Eugénia Vinagre, Isabel Praça, Zita Vale, Carlos Ramos

GECAD – Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development, Institute of Engineering, Polytechnic of Porto (ISEP/IPP), Portugal

{1090574, empvm, icp, zav, csr}@isep.ipp.pt

Abstract. The objective of this paper is to present a SMACK based platform for microgrids data storage and management. The platform is being used in a real microgrid, with an infrastructure that monitors and controls 3 buildings within the GECAD - ISEP/IPP campus, while, at the same time, receives and manages data sources coming from different types of buildings from associated partners, to whom intelligent services are being provided. Microgrid data comes in different formats, different rates and with an increasing volume, as the microgrid itself covers more customers and areas. Based on the atual available computational resources, a Big Data platform based on the SMACK stack was implemented and is presented. The Cassandra component of the stack has evolved. AC version 2 is still supported until the version 4 release, and is often still used in production environments. However, a new stable version, version 3, introduces major optimizations in the storage that bring disk space savings. The main focus of this work is on the Data Storage and the formalization of the data mapping in Cassandra version 3, which is contextualized by means of a short example with data coming from the monitoring infrastructure of the microgrid.

Keywords: Big Data Storage; Smart Grids; Cassandra

1 Introduction

Technological developments led to a huge spread of monitoring equipment that now provide an enormous quantity of data, based on which intelligent services may become available, turning into dynamic the traditionally centralized management of certain areas. In power systems, technological developments and the roll out of meters turn the Smart Grid (SG) as a new reality. Indeed, digital data sources range from sensors, that measure electric parameters (current, voltage, phase shift and frequency), to meters that monitor in real time consumption data and distributed generation sources, to environmental sensors (temperature, humidity, etc.), all of them being relevant to shift from a static structure to a more intelligent and flexible way to manage the electrical energy resources. The monitoring of the grid status results in a huge amount of collected data to deal and sharing with various parties [1].

The volume, velocity, and variety of the data make traditional data storage systems inappropriate to obtain the relevant value from the data analysis in a very short time.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_11

Big Data platforms are now the most promising way for the storing and analysis of high volumes of data. Apache technologies are being used in several domains. In this paper, we define a SMACK based architecture and implement a platform to support a real microgrid infrastructure existent at GECAD research group. Particular insights are given to the data storage process and the main focus of our contribution is given to the data mapping in relation to the partition size in Cassandra's most atual version.

The paper is structured into 4 sections, with section 2 addressing Big Data (BD) in SG context. Session 3 presents a platform based on SMACK, having Apache Cassandra (AC) as distributed storage for GECAD microgrid, with particular insights on the formalization of the data mapping process. Finally, the conclusions are presented in section 4.

2 Big Data and Smart Grids

To improve decision making, a system must be in place capable of collecting, managing, and processing information. In BD, the sheer volume of information requires new approaches when designing a solution that extracts knowledge within a reasonable period. This phenomenon, referred to as BD, is characterized by 5 Vs (i.e. Volume, Velocity, Variety, Veracity, Value) [2]. Each of these Vs represents real challenges (e.g. how to collect and transport a large volume of information; how to store this information, how to analyze and extract knowledge, how to ensure its security and privacy, how to process it in real time, etc.). The management of information with these characteristics raised great interest in the scientific and business community. Hadoop and Spark, are the most referenced frameworks.

The Apache Hadoop Framework was the first mainstream BD solution. It is based in Batch Processing, distributed file system HDFS (Hadoop Distributed File System), a programming model MapReduce and YARN (Yet Another Resource Negotiator) [3]. Apache Spark is a set of tools and high level APIs for large scale distributed processing of data in-memory [4]. Currently, Spark is considered as the most active open source project in BD. Its speed advantages, allied with an out of the box integration of data manipulation using SQL like syntax, support for several storage systems, and ability to distribute machine-learning computation, have contributed to its success.

In the new ecosystem of SG, all the players (i.e., power generation, transmission, distribution, customers, service providers, operations and markets) support their operations using a varied range of equipment that generate a large flow data. The last report issued by the European Union [5] refers numerous projects focusing the implementation of smart metering (SM). According to the same source, around 72 % EU customers are expected to be equipped with SM by 2020. The success of these projects launches an alert for the extensive amount of data generated in real time, that need adequate storage and analysis means to provide the development of dynamic services to better manage grid resources. Also, in the literature there are numerous references that characterize the type of data circulating on SG, at very high rates, as unstructured or, at most, semi-structured. Extrapolating this reality to the universe of the existent equipment and the foreseen roll outs by 2020 is easy to understand the

challenge is now on the data management. However, all the data being generated can only be transformed into value if properly analyzed in order to generate key knowledge in decision-making and the development of intelligent services able to dynamically manage the grid towards increasing sustainability, efficiency and safety. The value will be so much greater as the increasing ability to feed the ecosystem with data collected outside their own domain (e.g. atmospheric data, events, consumer behavior, etc.), correlate and analyses them, not only to decide and predict, but also to discover something even imaginable.

There are numerous operations in the SG area that require data analysis (e.g., operability; cybersecurity and privacy; self-healing fault; demand response; competitive energy markets; auto configuration; resource optimization; real-time decisions; forecasting; monitoring; etc.). Big data analytics is one of the biggest challenges in the BD domain. Traditional methodologies and algorithms are not prepared to run with large datasets (i.e. petabytes or more). Over the years, great efforts have been made on this issue and there are numerous references in the literature, of works and tools, for data analysis (based on batch and / or streaming) [4].

3 A Big Data Platform

Microgrids, sometimes referred as SG building blocks, are small areas with distributed energy resources that can operate in isolated mode. In this section we describe the BD cluster, with a distributed architecture, implemented in a real microgrid infrastructure existent in GECAD research centre [6].

3.1 GECAD microgrid

GECAD infrastructure includes 3 individual and independent buildings within the campus of the Institute of Engineering from the Polytechnic of Porto (ISEP/IPP), with photovoltaic (PV) and wind power generation; GECAD microgrid laboratory, which provides real-time simulation capabilities, a weather station from ISEP and consumption data from different types of buildings, some being monitored in real time and some others received through unstructured files. Fig. 1 illustrates the main data inputs for the platform.

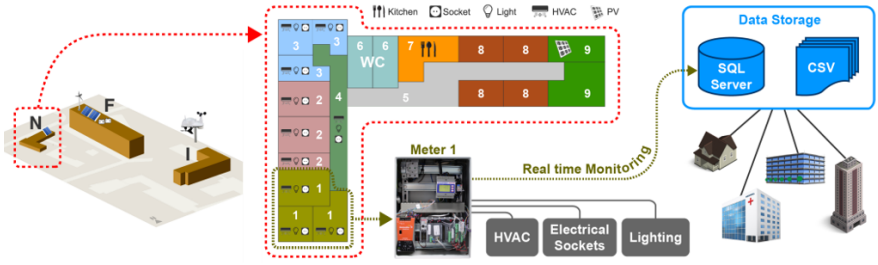


Fig. 1. Available data sources in GECAD's microgrid.

The 3 buildings are: Building F, with a PV system and a wind power system, where production is acquired every 10 seconds; a Building I, with consumption data acquisition every 10 to 15 seconds; and Building N, with its own PV system, injected directly into the building grid, consumption and generation is being acquired every 10 seconds. Details about GECAD microgrid can be found at [7].

Inside the buildings, three-phase energy analyzers measure data of three load groups grouped by rooms: Heating, Ventilation and Air Conditioning group (HVAC); Lighting group; and Electrical Sockets Group. The actual system stores data in time intervals ranging from 10 to 40 seconds (depending on the building) in a single SQL Server database.

To better illustrate the consumption data that is being acquired since 2014, with a total of 35 measures every 10 seconds, from grid frequency to current and voltage total harmonic distortion, power factors, apparent, active and reactive power, imported active energy, etc. Table 1 presents an excerpt of the information captured from the energy analyzers. The idea is to collect data directly from the energy analyzers, abandon the single server setup and store it in a distributed storage system.

Table. 1. Consumption data measurements.

TimeCol	...	N3_P1 (W)	N3_P2 (W)	N3_P3 (W)	...	N3_U1N (V)	N3_Q3 (VAr)	N3_PF3	...	N3_THD_U31
2014-07-24 09:57:30.0070		12,457793	373,495636	882,378357		236,062027	-47,430115	0,947596		1,931271
2014-07-24 09:57:40.0000		12,676662	376,461639	1224,399292		239,037048	28,984854	0,943464		1,942365
2014-07-24 09:57:50.0000		12,659256	374,636353	1279,258545		239,103882	18,341938	0,938458		1,940721

3.2 Distributed Architecture

A BD distributed architecture was designed and implemented in GECAd real microgrid. As illustrated in Figure 2, the design is based on the SMACK stack, a combination of the Lambda and Kappa architecture models, that uses primarily open-source technologies (Spark, Mesos, Akka, Cassandra and Kafka) in an orchestrated pipeline that tackles both batch and streaming analysis for real-time scenarios under a single development language (Scala or Java) [8].

The cluster's hardware resources and software are managed by Apache Mesos. Apache Kafka serves as a message queue, which also provides APIs for streaming data ingestion, from the SQL database and directly from the energy analyzers. AC serves as the distributed persistent storage database and enables application development, while Apache Spark provides a richer query language over the stored data and the creation of forecasting machine learning models and streaming analysis. In total, there are two nodes for storing data in AC and for Spark analysis. The master nodes' responsibility includes managing the slave resources, scheduling Spark applications and has a Kafka node for data ingestion and message queuing.

The system makes use of Kafka's Connect API to pull raw data from the energy analyzers through the Modbus communication protocol into Kafka, where it is stored temporarily. Finally, another connector moves data from Kafka maps it into Cassandra, where it is persisted indefinitely and available for random access reads.

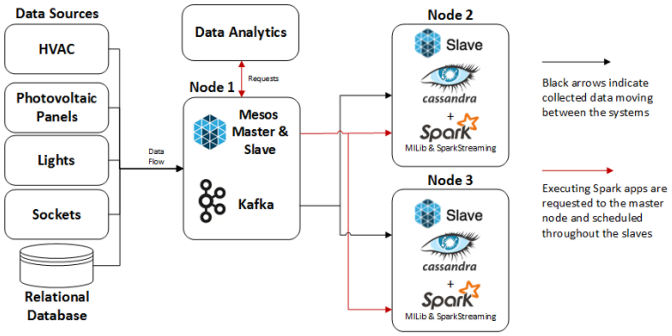


Fig. 2. Proposed BD architecture for GECAD's microgrid scenario.

This architecture is implemented with the computational resources actually available at GECAD. This serves as a first insight into the large-scale implementation of the architecture, with more advantageous computational resources and able to support several microgrids, e.g., a SG.

3.3 Data Storage in Cassandra

The actual SQL database presents itself as a serious storage system for massive data in the context of SGs. AC has been proposed to store historic data in a cloud-based architecture for SG [9] and considered to fulfill a similar role in a distributed data analytics platform for Wide-Area Synchrophasor Measurement Systems [10]. The system's success can be mainly attributed to its masterless architecture, linear scalability, multiple data center deployments and continuous availability.

However, due to scalability concerns, AC's Query Language (CQL) limits the possibilities of retrieving the data. When using this database, the queries are not an afterthought, but defined before the data model itself. In [11] the authors detail the concepts of the Cassandra data model and the preferred modeling methodology, including the concepts of: keyspaces, column families, partitions and clustering columns.

3.4 Data Mapping

Fig. 3 showcases a possible physical data model of a column family, illustrated with a small sub set of the registry, related to the active power of the 3 load groups from the energy analyzer N3, and its respective CQL statement. In this example, only the fields related to the load groups 'N3_P1', 'N3_P2' and 'N3_P3', from in Table. 1, are being used. A unique identifier is used as a partition key and the time of measurement is stored as a clustering column "datetime" in descending order.

In CQL, all selection statements must include the key. Retrieving all data at once can result in a timeout. Equality and lesser/greater searches on non-primary key columns are not possible, and as such, filtering records for any the load groups below or above a value is only possible, for instance, with SparkSQL.

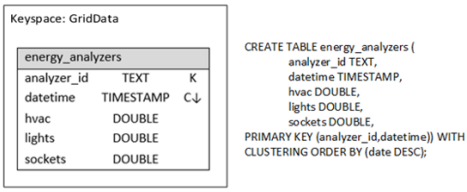


Fig. 3. Column Family for active power measurements and the respective CQL statement.

A visual representation of two energy analyzers partitions is seen in Fig. 4. The first energy analyzer has two rows separated in time while second only has a single record. From the image, we can see that the clustering value for the time of the measurement is repeated for each field in a row, and the field names are repeated across the whole partition. Both issues are addressed in AC version 3.

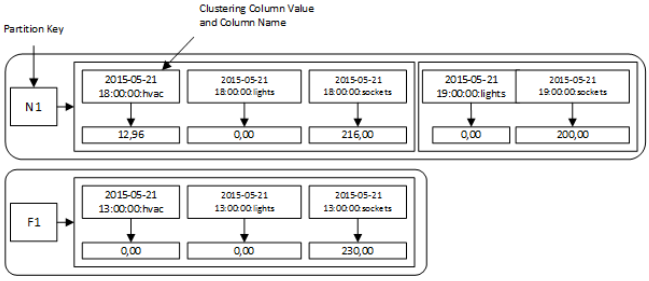


Fig. 4. Visual representation of how AC version 2 stores data on disk.

Since a single partition is stored in one cluster node, as the records grow, the likelihood of creating hot nodes in the cluster increases. Therefore, it is important to split partitions to improve cluster health and performance.

3.5 Partition Size

Cassandra has limitations on how wide a partition can grow in terms of the number of cells and its size, before suffering from performance concerns. The maximum number of cells is 2 billion and the recommended partition size is approximately hundreds of megabytes [12]. Because the data sources are infinite streams, the previously suggested partition key must be changed. In this case, the time bucketing method of splitting partitions is preferred. Time bucketing involves adding a new date or time field to the partition key, effectively splitting the partition into smaller groups in time. For instance, monthly partitions can be achieved using a concatenated string of the month and year. However, a single partition per query will, at most, return the data for a specific month and year. Retrieving more data is accomplished by issuing parallel statements.

Datastax's Academy Course DS220 presents two formulas for roughly estimating the partition size for a column family for AC version 2 [13]. Equation 1 states that the number of values (cells) N_v is equal to the product of the estimated number of rows

N_r with the number of regular columns (in parenthesis), plus the number of static columns N_s. Regular columns are a result of the subtraction of the number of primary key columns N_{pk} and static columns N_s to the total N_c.

$$N_v = N_r \times \left(N_c - N_{pk} - N_s\right) + N_s \ . \quad (1)$$

Given N_v, equation 2 is used to calculate the partition size P_s in bytes. The equation can be broken into parts: first, we add the summations of the sizes of primary key columns C_k and static columnsC_s; second, multiply N_r with the addition of the size of each regular column C_r and the total size of all clustering columns C_c (Cassandra repeats the clustering values for each cell); third, we multiply the value of 8 (size of a hidden timestamp hidden in each cell)) to previously obtained N_v.

$$P_s = \sum_i sizeOf(C_{k_i}) + \sum_j sizeOf(C_{k_j}) + Nr \times \sum_k \left(sizeOf(C_{r_k}) + \sum_l sizeOf(C_{c_l}) \right) + 8 \times N_v \, . \quad (2)$$

If the data collection occurs at every second, at the end of a week, the value for N_r is 604 800 and $N_v = 604\,800 \times (5-2)$ totaling 1 814 400. For the partition size, we first assume the size in bytes of each of the fields: 'analyzer_id' 10 bytes; 'datetime' 8 bytes; 'hvac' 'lights' and 'socket's all 8 bytes each. Thus, $P_s = 10 + 0 + 604\,800 \times \left((8+8) + (8+8) + (8+8)\right) + 8 \times 1\,814\,400$, totaling approximately 42 MB.

Table 2 displays the results for various time resolutions and partition sizes at the end of the day, week and month. With the time resolution of one second, a monthly partition can be used cautiously, while weekly partitions keep the size below 100 MBs. With the time resolution of one second, a monthly partition can be used cautiously, while weekly partitions keep the size below 100 MB.

Table 2. Partition Size for one energy analyzer with different time resolutions for AC v2.

Time Resolution	End of	N_r	Nv	P_s (MB) Approximately
Each second	Day	86400	259 200	6
	Week	604 800	1 814 400	42
	Month	2 592 000	7 776 000	178
10 seconds	Day	288	864	0,6
	Week	2016	6048	4,1
	Month	8640	25 920	18

The storage engine for version 3.0 changes radically, providing storage savings [14]. The major differences include: the timestamps for conflict resolution are delta-encoded and can be written only once per row when all the cells have the same timestamp; the field names are stored at a row level; and the clustering column values are no longer repeated for each cell. Other important changes include better serialization that is discussed in detail in [15].

As of the latest version of AC (i.e. version 3.0), there are no widely accepted formulas for estimating partition size. We propose to change equation 2 to:

$$P_s = \sum_i sizeOf\left(C_{k_i}\right) + \sum_j sizeOf\left(C_{s_j}\right) + Nr \times \left(\sum_k sizeOf\left(C_{r_k}\right) + \sum_l sizeOf\left(C_{c_l}\right) + 8\right). \quad (3)$$

Equation 3 moves the clustering columns and the hidden timestamp to a row level. It is important to note that both equations do not consider the field names serialization of the different storage engines. This is important when using columns with dozens of regular columns, as the effects in size will be more significant. The updated results can be seen in Table 3. There is a substantial decrease in partition size using the new storage engine, and the monthly partition can be recommended.

Table 3. Partition size for one energy analyser using the adapted formula.

Time Resolution	End of	N_r	Nv	P_s (MB) Approximately
Each second	Day	86400	259 200	3,2
	Week	604 800	1 814 400	23
	Month	2 592 000	7 776 000	98

It is important to remember that partition bucketing will have consequences on the querying strategy. The best approach is to consider the widest partition strategy relative to the application requirements while trying to maintain the size to 100 MB or below.

4 Conclusions

In this paper a BD platform for SG has been proposed. It is based on Smart Stack architecture, implemented in a distributed way and already being fed by several data sources from GECAD microgrid infrastructure and external sources from partners. The implementation was done with the available computational resources at GECAD, but this cluster provides a first insight about the scale up of the platform for SG management. An important contribution is the data mapping in AC version 3.0 and how the partition size should be done to take the best profit of the data for real time data analysis. It's critical to have the ability to compare multiple data stores intelligently and objectively so that sound architectural decisions can be made. So, in the next step we will validate the platform proposed with YCSB (i.e. an open standard for comparative performance evaluation of NoSQL data stores). Further expanding the cluster with additional computational resources is something previewed in the short run.

Acknowledgements. This work has received funding from EU Horizon 2020 R&D programme under Marie Sklodowska-Curie grant agreement No 641794 (project DREAM-GO), EUREKA - ITEA2 Project FUSE-IT (nr.13023), Project GREEDI

(ANI|P2020 17822), FEDER Funds through COMPETE program and National Funds FCT under the UID/EEA/00760/2013 and SFRH/BD/103089/2014.

References

1. Skopik, F., Ma, Z.: Attack Vectors to Metering Data in Smart Grids under Security Constraints. In: 2012 IEEE 36th Annual Computer Software and Applications Conference Workshops. pp. 134–139. IEEE (2012).
2. Chandarana, P., Vijayalakshmi, M.: Big Data analytics frameworks. In: 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA). pp. 430–434. IEEE (2014).
3. Shvachko, K., Kuang, H., Radia, S., Chansler, R.: The Hadoop Distributed File System. In: 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST). pp. 1–10. IEEE (2010).
4. Zaharia, M., Franklin, M.J., Ghodsi, A., Gonzalez, J., Shenker, S., Stoica, I., Xin, R.S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S.: Apache Spark: A Unified Engine for Big Data Processing. Commun. ACM. 59, 56–65 (2016).
5. Felix Covrig, C., Ardelean, M., Vasiljevska, J., Mengolini, A., Fulli, G., Amoiralis, E., Sánchez Jiménez, M., Filiou, C., European Commission. Joint Research Centre. Institute for Energy and Transport.: Smart grid projects outlook 2014. Publications Office (2014).
6. Vinagre, E., Gomes, L., Vale, Z.: Electrical energy consumption forecast using external facility data. Proc. - 2015 IEEE Symp. Ser. Comput. Intell. SSCI 2015. 659–664 (2016).
7. Gomes, L., Lefrancois, M., Faria, P., Vale, Z.: Publishing real-time microgrid consumption data on the web of Linked Data. In: 2016 Clemson University Power Systems Conference (PSC). pp. 1–8. IEEE (2016).
8. Estrada, R., Ruiz, I.: Big Data SMACK. Apress, Berkeley, CA (2016).
9. Mayilvaganan, M., Sabitha, M.: A cloud-based architecture for Big-Data analytics in smart grid: A proposal. In: 2013 IEEE International Conference on Computational Intelligence and Computing Research. pp. 1–4. IEEE (2013).
10. Zhou, D., Guo, J., Zhang, Y., Chai, J., Liu, H., Liu, Y., Huang, C., Gui, X., Liu, Y.: Distributed Data Analytics Platform for Wide-Area Synchrophasor Measurement Systems. IEEE Trans. Smart Grid. 1–9 (2016).
11. Chebotko, A., Kashlev, A., Lu, S.: A Big Data Modeling Methodology for Apache Cassandra. In: 2015 IEEE International Congress on Big Data. pp. 238–245. IEEE (2015).
12. Datastax: Estimating partition size, https://docs.datastax.com/en/landing_page/doc/landing_page/planning/planningPartitionSize.html. Accessed 23 Nov 2016
13. Datastax Academy: Physical: Partition Size, https://academy.datastax.com/courses/ds220-data-modeling/physical-partition-size. Accessed 20 Nov 2016
14. Lebresne, S.: Putting some structure in the storage engine, http://www.datastax.com/2015/12/storage-engine-30. Accessed 4 Dec 2016
15. Morton, A.: Introduction To The Apache Cassandra 3.x Storage Engine, http://thelastpickle.com/blog/2016/03/04/introductiont-to-the-apache-cassandra-3-storage-engine.html. Accessed 4 Dec 2016

Data Mining for Prosumers Aggregation considering the Self-Generation

Catarina Ribeiro[1,2], Tiago Pinto[1], Zita Vale[1], José Baptista[2]

[1]GECAD – Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development, Institute of Engineering, Polytechnic of Porto (ISEP/IPP), Portugal
[2]UTAD – Universidade de Trás-os-Montes e Alto-Douro
{acrib, tmcfp, zav}@isep.ipp.pt, baptista@utad.pt

Abstract. Several challenges arrive with electrical power restructuring, liberalized electricity markets emerge, aiming to improve the system's efficiency while offering new economic solutions. Privatization and liberalization of previously nationally owned systems are examples of the transformations that have been applied. Microgrids and smart grids emerge and new business models able to cope with new opportunities start being developed. New types of players appear, allowing aggregating a diversity of entities, e.g. generation, storage, electric vehicles, and consumers, Virtual Power Players (VPPs) are a new type of player that allows aggregating a diversity of players to facilitate their participation in the electricity markets. A major task of VPPs is the remuneration of generation and services (maintenance, market operation costs and energy reserves), as well as charging energy consumption. The paper proposes a normalization method that supports a clustering methodology for the remuneration and tariffs definition. This model uses a clustering algorithm, applied on normalized load values, the value of the micro production, generated in the bus associated to the same load, was subtracted from the value of the consumption of that load. This calculation is performed in a real smart grid on buses with associated micro production. This allows the creation of sub-groups of data according to their correlations. The clustering process is evaluated so that the number of data sub-groups that brings the most added value for the decision making process is found, according to players characteristics.

1 Introduction

Power sector has been completely revolutionized by the emergence of liberalized electricity markets (EM) aiming to improve the system's efficiency while offering economic solutions. Characterized by an increase in competition and changes in participant entities, potential benefits will depend on the efficient operation in the market and in bilateral contract negotiation and remuneration of aggregated players. In consequence of these structural changes, has been a gradual decentralization of decision making and a redistribution of responsibilities for different players [1]. Important developments concerning electricity market players modelling and simulation including decision-support capabilities can be widely found in the literature [2-3]. Subsystems of the main network are evolving into a reality. The coordination of

The present work has been developed under the EUREKA - ITEA2 Project FUSE-IT (ITEA-13023), Project GREEDI (ANI|P2020 17822), and has received funding from FEDER Funds through COMPETE program and from National Funds through FCT under the project UID/EEA/00760/2013.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_12

these entities is a challenge that requires the implementation of distributed intelligence, potentiating the concept of Smart Grid (SG) [4, 5]. However, EM and SG are not converging towards common goals and technical and economic relationships are addressed in an over simplistic way. Operation methods and EM models do not take all the advantage of installed DG, yielding to inefficient resource management that should be overcome by adequate optimization methods [6]. The aggregating strategies of players, allow them to gain technical and commercial advantages, individuals can achieve higher profits due to specific advantages of a mix of technologies to overcome disadvantages of some technologies. The aggregation of players gives rise to the concept of Virtual Power Player (VPP) [7], heterogeneous entities that aggregate different types of resources where each aggregated player has its individual goals. The VPP should conciliate all players in a common strategy, while enabling each player to pursue its own objectives [8]. There are some simulators in literature that enable modeling VPPs aggregation and resource management process. One relevant system in this domain is MASGriP (Multi-Agent Smart Grid Simulation Platform) [9], manages and controls the most relevant players acting in a SG environment. MASGriP is connected to MASCEM (Multi-Agent Simulator for Competitive Electricity Markets) [2,7], thus providing the means to perform joint simulations. A decision support system has been integrated in MASCEM, in order to allow players to automatically adapt their strategic behavior according to the operation context and with their own goals. This platform is Adaptive Learning Strategic Bidding System (ABidS) [10], and it provides agents with the ability of analyzing contexts of negotiation. This paper proposes a data mining methodology, based on the application of a clustering algorithm, applied on normalized load values, which groups the typical load profile of the consumers of a SG according to their similarity for the remuneration and tariffs definition from VPPs. The value of the micro production, generated in the bus associated to the same load, was subtracted from the value of the consumption of that load. This allows the creation of sub-groups of data according to their correlations. The clustering process is evaluated so that the number of data subgroups that brings the most added value for the decision making process is found,

2 RemT – Remuneration and tariff decision support tool for electricity markets

The Remuneration and Tariff Mechanism (RemT) [11, 12] is a decision support mechanism that is being developed to support the VPP actions in the definition of the best tariff and remuneration to apply to each of the aggregated players, regarding the VPP objectives and the individual goal of each aggregated player. In the scope of MASCEM, VPPs use RemT to remunerate aggregated players, according to the results obtained in the electricity market, the penalties for breach of contract, contracts established to guarantee reserve, demand response programs and incomes of aggregated consumers. The establishment of remuneration and tariffs is based on the identification of players' types and development of contract models for each player type. This considers players with a diversity of resources and requirements, playing several distinct roles (consumers, producers and can be responsible for one or several electrical vehicles). The players modelling takes into account the operation and market context. The terms for new contracts and best strategies for each context are determined by means of machine learning methods and data-mining algorithms. The definition process is presented in Figure 1.

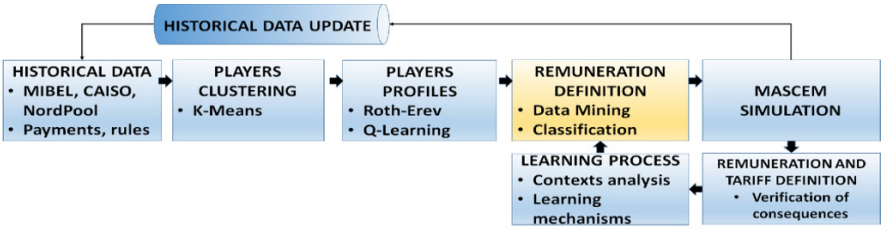


Figure 1. RemT definition process.

A. Clustering approach

There are many clustering algorithms in literature and unfortunately, no single algorithm that can by itself, discover all sorts of cluster shapes and structure[13]. K-means[14], has been used, as it proves to be a robust model for distinct applications: it minimizes distance from each point to the centre of the respective cluster, where μ_i is the mean of points in C_i, *i.e.* the cluster *centroid* as defined in (1).

$$\min \sum_{i=1}^{k} \sum_{x \in c_i} \left\| x - \mu_i \right\|^2 \tag{1}$$

To determine the quality of the division of players into different clusters the clusters validity indices MIA and CDI[15] were used, as formalized in (2) and (3).

$$MIA = \sqrt{\frac{1}{K} \sum_{k=1}^{K} d^2 (x^{(k)}, \mu^{(k)})} \tag{2}$$

$$CDI = \frac{\sqrt{\frac{1}{K} \sum_{k=1}^{K} \left[\frac{1}{2n^{(k)}} \sum_{n=1}^{n^{(k)}} d^2 (x^{(m)}, \mu^{(k)}) \right]}}{\sqrt{\frac{1}{2K} \sum_{k=1}^{K} d^2 (x^{(k)}, R)}} \tag{3}$$

Where d represents the Euclidian distance between two points, and R is the representative load profile of all consumers. This indices represent distances, the smaller (or greater) is the MIA and CDI value, it indicates more (or less) compact clusters. To facilitate the analysis of results, we will consider that the higher the value of the MIA and CDI, the larger the error associated with this cluster.

B. Normalization method considering prosumers self-generation

Analysing the results of previous work [11], is possible to conclude that aggregation strategies have very good results, and are very useful, because they provide a good separation according to what is intended. The non-normalization grouping process has led to a clear separation between different consumers types, as it considers the absolute consumption amounts in the clustering process. The normalized data, used as formalized in (4) and (5), reveals a separation through consumption profiles, although it is not able to consider the differences in consumption quantity. L is the value of load.

$$N_{c,h} = \frac{L_{c,h}}{ML_c}, \forall c \in co \tag{4}$$

$$ML_c = \max(L_c), \forall c \in co \tag{5}$$

Where *N* is the common normalized load, for each consumer *c*, for each hour *h, a*nd *co* is the set of all considered consumers. *ML* is the largest consumption value, of the consumer *c*, considering all hours. To improve the results achieved in the previous works, the difference normalization process is introduced. In different process the value of the micro production *P*, generated in the bus associated to a load, was subtracted from the value of the consumption of that load. This calculation is performed for loads from 1 to 17, loads which are on buses with associated micro production, for each specific load in the 24 periods. It is formalized in (6) and (7).

$$T_{c,h} = \left| L_{c,h} - P_{c,h} \right|, \forall c \in co \tag{6}$$

Where T corresponds to the load where consumption subtracted by micro production, for each consumer c, for each hour h.

$$SN_{c,h} = \frac{T_{c,h}}{SML_h}, \forall c \in co \tag{7}$$

Where *SN* is the load with a different normalization process, for each consumer *c*, for each hour *h*. *SML* is the largest consumption value recorded for all consumers at the time *h*. The values of production in each bus were taken on site, in the real distribution network [16]. This method aims to combine the advantages of both previous approaches (using non-normalized data, and regular normalization subtracting the value of production of the different loads), so as to achieve consumer groups that capture both differences in the quantities of consumption and also the trends of consumer profiles along the hours. Clustering process takes into account the tendency of the consumption values trough the time, regardless of its absolute amount. This separation is very important, according to different consumers´ types and profiles, it works as a base for personalized and dynamic consumption tariff definition. This approach ensures that data are also normalized in a range between 0 and 1, but without losing information related to differences between amounts of consumption among consumers. Two independent variables are subtracted to accentuate the difference between classes of loads. This is because, apparently, local production depends on the class of loads, see [16], residential houses produce more throughout the year than they consume whereas it is the opposite for commercial building, and residential building produce as much as they consume. This supports the use of the proposed method. While using the regular normalization, the value 1 is attributed to the greater consumption value of each consumer (thus both consumers with large and small values will always have one value of 1 in a certain hour), using the customized normalization method, only the largest consumer of all, will have a value of 1. The smaller consumers have normalized values with smaller values, proportional to the difference between the consumption quantities of that consumer and the largest consumer in each hour. Thus normalization is still made between 0 and

1, but there is visible difference between higher and lower consumption among different consumers, and the evolution of consumption of each consumer profile.

4 Case study

This case study intends to show the adequacy of the proposed normalization clustering methodology to solve the problem of remuneration of players with heterogeneous characteristics and behaviors. In order to test the adequacy of the method, a clustering algorithm has been applied, concerning the consumption data of a total of 82 consumers (8 residential houses, 8 residential buildings with 72 loads, and 2 commercial buildings). Data has been collected from a real distribution network throughout one year. The Smart grid accommodates distributed generation (photovoltaic and wind based generation) and storage units, which are integrated in the consumption buildings. The accommodated photovoltaic generation, wind based generation and storage units are related to the building installed consumption power, according to the current legislation in Portugal. Further details on the considered distributed network can be seen in [16]. The K-means algorithm has been used to perform the clustering process using non-normalized values of load (section A), and also normalized values, using both the regular normalization method (section B) and the proposed customized normalization method (section C).

A. Non-normalized data

The clustering process is performed for different numbers of clusters, in order to enable grouping consumers according to the similarity of their consumption profiles, in order to support the definition of specific tariffs that are suited for each of the consumer groups. From [13] it has been concluded that, by analyzing MIA and CDI results from the clustering of non-normalized data, the best clustering results are achieved with the use of 3 clusters, as the clustering error is minimal. When using 2 clusters, a clear separation of residential houses and buildings from commercial buildings is visible. It is also visible that the two commercial buildings (corresponding to loads 1 and 2) have been allocated to cluster 1, and the rest of the loads, corresponding to residential consumers, have been aggregated in cluster 2. This can be observed in Figure 2 which presents the load profiles of consumers that have been grouped in cluster 1 and in cluster 2 using the non-normalized data.

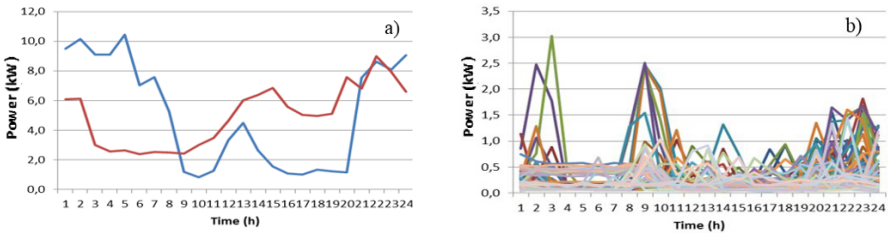


Figure 2. Consumption profile of loads allocated to: a) cluster 1; b) cluster 2

From Figure 2 it is visible that cluster 1 includes the two commercial buildings, with very distinct load profiles, and cluster 2 includes all the residential buildings and houses. When considering the grouping process with 3 clusters, the difference is that there is still a separation from residential houses and buildings to the commerce.

However, in this case the two types of commercial buildings are also separated, as they present very different load profiles.

B. Normalized data

In the second clustering process regular normalized data were used. Normalization was made considering each type of consumer. The value of load corresponding to each period was divided for the maximum value register in that specific load in the 24 periods. When using normalized values, a more accentuated descent of the clustering error values is visible. The descent in the error value is stable from the start, which hardens the identification of the optimal number of clusters that should be used. For this reason it is not advantageous to use more than 2 or 3 clusters, since the use of a larger number is not reflected by a significant gain in clustering error. Analyzing the results with 2 clusters, the separation is not as clear as it was with non-normalized values. The two commercial buildings(load 1 and 2), were aggregated in different clusters, with residential consumers. However, the clustering process with normalized values has better results from the load profile separation stand point. Figure 3 presents the allocation of the consumers considering normalized and 2 clusters.

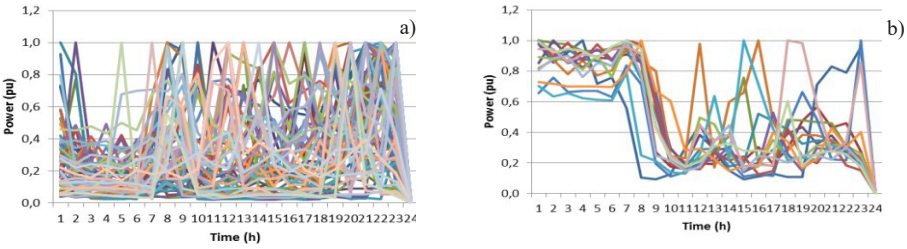


Figure 3. Consumption profile of loads represented in: a) cluster 1, b) cluster 2

From Figure 3 is visible that although the consumer types cannot be separated correctly with this approach as occurs when using non-normalized data (Figure 2), the separation of the load profiles is more evident in this case, since profiles are grouped independently from the gross amount of consumption itself.

C. Proposed normalization process

In this process, the value of the micro production, generated in the bus associated to the same load, was subtracted from the value of the consumption of that load. This calculation is performed for loads from 1 to 17, these are the loads, which are on buses with associated micro production. MIA and CDI are used to analyze the clustering error, they are presented in Figure 4.

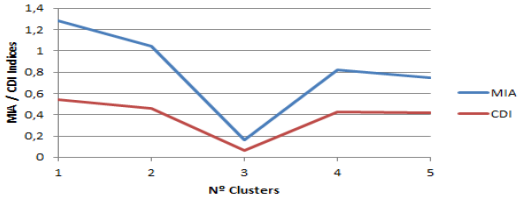


Figure 4. MIA and CDI results for difference normalization values for 24 hour period.

In difference normalization process, the best clustering results are achieved with the use of 3 clusters, as the clustering error is minimal, similar to what happened in previous cases. Using 2 clusters, a clear separation of residential houses and buildings from commercial buildings is visible. Also, the two commercial buildings (corresponding to loads 1 and 2) have been allocated to cluster 1, and the rest of the loads, corresponding to residential consumers, have been aggregated in cluster 2. When considering the grouping process with 3 clusters, the difference is that there is an even better separation of consumers types, commercial buildings were allocated to cluster 1, residential houses to cluster 2 and residential building to cluster 3. In this case, the two types of commercial buildings stayed in the same cluster, although they present very different load profiles.

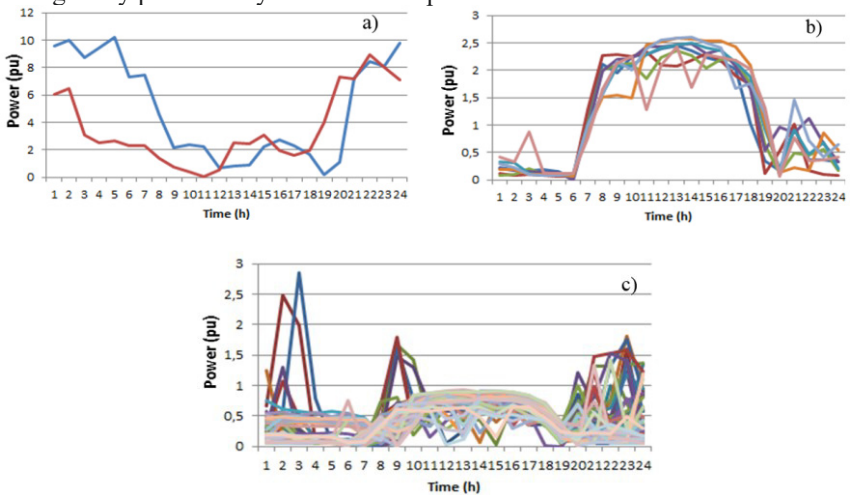


Figure 5. Consumption profile of load allocated to: a) cluster 1; b) cluster 2; c) cluster 3

The proposed difference normalization brings, therefore, clear advantages to the RemT tariff definition process. It enables to clearly identify different consumers, taking into account their consumption tendency and amount, therefore breaking the way for an objective and fare definition of dynamic electricity tariffs, which can suitably fit each of the identified groups, i.e. consumers with similar consumption tendencies, taking into account their dimension. The new type of normalization, when compared with the previous normalization types, allows an even clearer separation of consumer types, which is evident from the load profile graphs that show the separation into different clusters, and also by the MIA and CDI values, which show that the proposed method achieves smaller clustering error values than the other methods.

6 Conclusions

EMs are experiencing profound transformations. This case study demonstrated the usefulness and advantage of data mining methodologies, based on the application of clustering process to group typical load profiles of consumers according to their similarity to allow proposing specific consumption tariffs to each group, so that consumers load profile is taken into account to meet the objectives of the SG aggregator. This work allows the development of a tool that provides a decision support for VPP

definition of best tariff and remuneration to apply to each aggregated player, RemT. To develop RemT a clustering methodology that uses different data normalization methods was presented, and a new difference normalization method has been introduced. The results of the presented case study, based on real consumption data, show that the difference normalization method combines the advantages of both previous approaches, so as to achieve more consumer groups that capture both differences in the quantities of consumption, as well as the trends of consumer profiles along hours. This is crucial, according to different consumers´ types and profiles, as it works as a basis for personalized and dynamic consumption tariff definition. Thus normalization is the same made between 0 and 1, but there is visible difference between higher and lower consumption among different consumers, and the evolution of consumption of each consumer profile is also captured. RemT mechanism is evolving to become a crucial tool to go a step forward in EM simulation, by enabling a fair and dynamic means to define electricity tariffs for different types of consumers.

References

1. L. Meeus, et al., “Development of the Internal Electricity Market in Europe”, The Electricity Journal, vol. 18, no. 6, pp. 25-35, 2005
2. I. Praça, C. Ramos, Z. Vale, M. Cordeiro, “MASCEM: A Multi-Agent System that Simulates Competitive Electricity Markets”, IEEE Int. Systems, 18,6,54-60, 2003
3. V. Koritarov, "Real-World Market Representation with Agents: Modeling the Electricity Market as a Complex Adaptive System with an Agent-Based Approach", IEEE Power & Energy magazine, pp. 39-46, 2004
4. M. Shahidehpour, et al., “Market Operations in Electric Power Systems: Forecasting, Scheduling, and Risk Management”, Wiley-IEEE Press, pp. 233-274, 2002
5. Blumsack S and Fernandez A. “Ready or not, here comes the smart grid!” Energy. 2012; 37(1):61-8
6. Sousa, T. et al., "Intelligent Energy Resource Management Considering Vehicle-to-Grid: A Simulated Annealing Approach," IEEE Trans. on Smart Grid, 3, 535-542, 2012
7. Z. Vale, T. Pinto, I. Praça, H. Morais, "MASCEM - Electricity markets simulation with strategically acting players", IEEE Intelligent Systems, vol. 26, n. 2, Special Issue on AI in Power Systems and Energy Markets, 2011
8. T. Pinto, et al, “Multi-Agent Based Electricity Market Simulator With VPP: Conceptual and Implementation Issues”, 2009 IEEE PES General Meeting, 2009
9. Oliveira, P. et.al., "MASGriP - A Multi-Agent Smart Grid Simulation Plataform," IEEE 2012 - Power and Energy Society General Meeting, San Diego, USA, 2012, pp. 1-10
10. Pinto, T., et.al., “Adaptive Learning in Agents Behaviour : a Framework for Electricity Markets Simulation,” Integr. Comput. Aided. Eng., vol. 21, no. 4, pp. 399–415, 2014
11. C. Ribeiro., et al., “Data Mining approach for Decision Support in real data based Smart Grid scenario” IATEM, 2015
12. Ribeiro C., et al., “ Intelligent Remuneration and Tariffs in for Virtual Power Players”, IEEE PowerTech (POWERTECH) Grenoble, France, 16-20 June, 2013
13. Anil K. Jain et. al., (1999) “Data Clustering: A Review.” ACM Computing Surveys, 31 (3). pp. 264-323.
14. Anil K. Jain, “Data Clustering: 50 years beyond K-Means”. Pattern Recognition Letters, Elsevier, Vol. 31, Issue 8, pp.651-666, June 2010.
15. Chicco et al., “Support Vector Clustering of Electrical Load Pattern Data”. IEEE Transactions on Power Systems, vol.24, no.3, pp.1619-1628, August 2009.
16. Canizes B. et. al., “Resource Scheduling in Residential Microgrids Considering Energy Selling to External Players”, Power Systems Conference (PSC 2015), South Carolina, USA, 10-13 March, 2015

Part III
Distributed Computing and Artificial Intelligence

Control of Accuracy of Forming Elastic-Deformable Shafts with Low Rigidity

Antoni Świć[1], Arkadiusz Gola[1,*],

[1] Institute of Technological Systems of Information, Faculty of Mechanical Engineering, Lublin University of Technology, Poland
{a.swic,a.gola}@pollub.pl

Abstract. The paper presents the problem of control of the accuracy of forming elastic-deformable shafts with low rigidity and namely the control of the elastic-deformable state of semi-finished product with low rigidity through the effect of additional force factors under conditions of lateral bending, permitting the achievement of uniform stiffness of the shaft at the point of application of machining force, and consequently a significant increase of the accuracy of shaft formation in the course of machining.

Keywords: shafts, low rigidity, forming, machining, accuracy, control.

1 Introduction

Analysis of industrial products shows that shafts with low rigidity are more and more frequently used in the machines and mechanisms produced. In the machine building industry they account for approximately 12% of axial-symmetrical parts that constitute 34% of all parts produced [1,2].

Parts with low rigidity find application in the aerospace industry (flexible, elastic and torsion shafts, springs, bolts), tooling industry (jigs and fixtures of various kinds, mechanisms, precision and special tools, drill bits, reamers, screw taps, boring bars), machine building industry (shafts, turbine and pump rotors, feed screws), agricultural machines (tractor and combine harvester axle shafts), in the motor industry and related industries.

The problem of machining low-rigidity shafts was widely discussed in the literature. There are many publications concerning the problem of stability lobes [3-5], definition of cutting conditions [6-7], machining forces and error prediction [8-9] or chatter suppression [10]. Moreover, the problem of simulation of low-rigidity part machining was in the special interests of researchers [11-14]. An important issue in the production of such elements of machines is the achievement of the required accuracy of their form and dimensions. Unfortunately, it is rather difficult to find the papers concerning the problem of accuracy control during elastic-deformable shafts forming.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_13

The main cause of errors in the longitudinal section of elastic-deformable shafts with low rigidity occurring in the course of machining is non-uniform stiffness of the technological system along X axis at the point of action of machining force [15-18]. The form error of the machined surface can be reduced by ensuring uniform stiffness of the technological systems at various sections of the semi-finished product (at the point of application of machining force).

The paper presents the problem of control of the accuracy of forming elastic-deformable shafts with low rigidity, and namely the control of the elastic-deformable state of semi-finished product with low rigidity through the effect of additional force factors under conditions of lateral bending, permitting the achievement of uniform stiffness of the shaft at the point of application of machining force, and consequently a significant increase of the accuracy of shaft formation in the course of machining.

2 Modelling of accuracy of forming of shafts with low rigidity

The value of deformation in the machining zone (under the turning tool) was adopted as the minimised function in the control of accuracy of forming elastic-deformable shaft with low rigidity. A schematic of forming of a shaft with low rigidity is presented in Fig. 1.

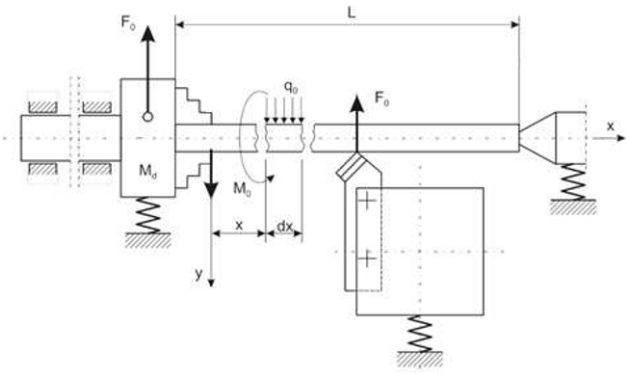


Fig. 1. A general schematic of low-rigidity shaft turning

It was assumed that deflection in a section with coordinate $a = x$ from the thrust component of machining force F_p and from additional force F_d equals zero [19,20].

$$y(a) = 0 \tag{1}$$

In the control of machining accuracy of parts in the elastic deformable state, in the case of cross-bending the use of lever-type vibration dampers [20] leads to the calculation scheme [21,22]. On the basis of the equation of deflections and function (1) we obtained relations for the control of lever-type vibration damper.

$$F_d(x) = \frac{F_p x\left[L^2(4L-9x)+x^2(6L-x)\right]}{\left\{2L^3\left[3(x+\delta)-x\right]-(x+\delta)^2(3L-x)\left[3L-(x+\delta)\right]\right\}} \tag{2}$$

where: F_d – additional force, F_p – component of machining force, L – length of the machined shaft, δ - distance between the point of application of machining force and the additional force.

and with the inclusion of M_1

$$F_d(x) = \frac{F_p\left\{x\left[L^2(4L-9x)+x^2(6L-x)\right]+L\left[6L^2(3x-L)-3x^2(5x-L)\right]/30\right\}}{\left\{2L^3\left[3(x+\delta)-x\right]-(x+\delta)^2(3L-x)\left[3L-(x+\delta)\right]\right\}} \tag{3}$$

Where: at $\delta = 0$, $M_1 = 0$, $x = 0$: $F_d(0) = 0$ and at x = L: $F_d(L) = 1$.

Graph of the relation F_d/F_p (x/L) at $\delta = 0$, $M_1 = 0$, is presented in Fig. 2b. Two cases were considered – when force F_d is applied beyond the machining zone $(x+\delta)$ and before the machining zone $(x+\delta)$. Analysis of the obtained analytical relations shows that moving the vibration damper to the left or to the right from the machining zone leads to an increase of force F_d with compensation of bending deformations from the forces and moments of machining.

Control of the elastic-deformable state as a result of application of bending moments to the faces of the part is used in the control of forming accuracy in oscillation grinding of shafts with low rigidity. As the objective functions of control we can adopt the minimum of linear and angular deformations (deflection and angle of rotation) in the zone of contact of the tool with the part [1,3,23]. The operation of grinding of shafts with low rigidity the total deflection of the part $y(x)$ under the effect of the bending force F_p and moments M_1 and M_2 in the section with coordinate x (under the tool), based on the principle of superposition, in the case of linear systems we can express as:

$$y(x) = y_1(x) + y_2(x) + y_3(x), \tag{4}$$

and the total angle of rotation $y(x)$ of part section with coordinate x is determined as the first derivative of deflection

$$y'(x) = y_1'(x) + y_2'(x) + y_3'(x), \tag{5}$$

where: $y_1(x)$, $y_1(x)$ - deflection and angle of rotation, respectively, in part section with coordinate x as a result of elastic deformations of parts with low rigidity; $y_2(x)$, $y_2(x)$ - as a result of deformation of jaws; $y_3(x)$, $y_3(x)$ - as a result of initial displacements caused by geometric inaccuracy of alignment of the jaws and the part.

In all the cases the calculation of deflections and angles of rotation was performed with the use of Mohr's integral (acc. to the Mohr-Vereshchagin rule) with verification with the method of initial parameters.

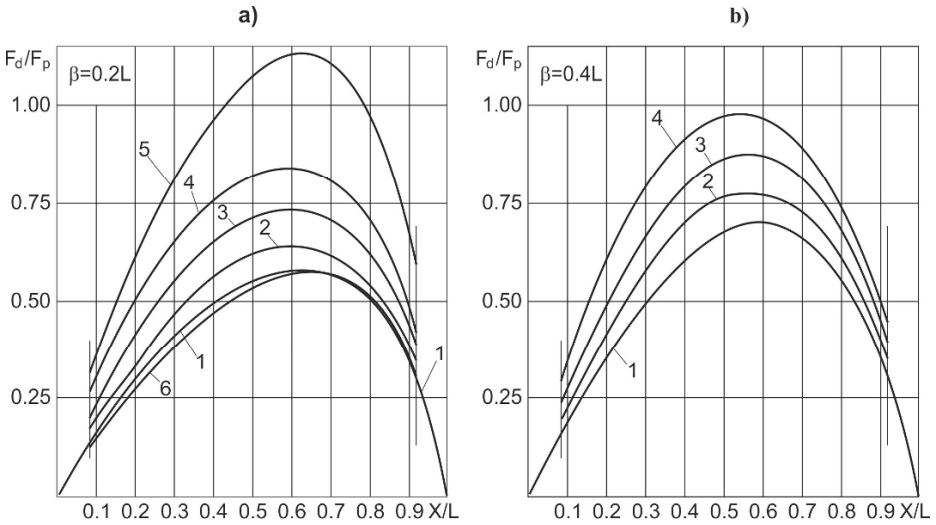


Fig. 2. Relation of change of F_d/F_p in the case of twin-stop dampers at: a) $\beta = 0{,}2L$; b) $B = 0{,}4\ L$

Taking into account the adopted objective function, relations (4) and (5) are compared to zero. After transformation we obtained a system of linear algebraic equations:

$$\left.\begin{aligned} A_1 M_1(x) + A_2 M_2(x) &= B_1 \\ A_3 M_1(x) + A_4 M_2(x) &= B_2 \end{aligned}\right\} \tag{6}$$

From the system of equations (6) we obtained the relations of control of bending moments $M_1(x)$ and $M_2(x)$, that guarantee the minimum of bending deformations in the machining zone with the inclusion of the stiffness of the stops and of the geometric inaccuracy of alignment:

$$M_1(x) = (B_1 A_4 - B_2 A_2)/(A_1 A_4 = A_2 A_3) \tag{7}$$

$$M_2(x) = (B_2 A_1 - B_1 A_3)/(A_1 A_4 - A_2 A_3) \tag{8}$$

where the following denotations were adopted:

$$A_1 = -\left[x(L-x)/6L^2\right]\left[x^2 + 3x(L-x) + 2(L-x)^2\right] + m^3(L-2x)/3n,$$

$$A_2 = -\left[x(L-x)/6L^2\right]\left[2x^2 + 3x(L-x) + (L-x)^2\right] + m^3(2x-L)/3n,$$

$$A_3 = \left(1/6L^2\right)\left(6xL^2 - 3x^2L - 2L^3\right) - 2m^3/3n,$$

$$A_4 = \left(1/6L^2\right)\left(3x^2L - L^3\right) + 2m^3L/3n,$$

$$B_1 = -\left[x(L-x)/6L^2\right]\left\{2F_p\left[x^2(L-x) + L - x^2\right]\right\} + \left(m^3L/3n\right)x$$

$$\left[F_p\left(L^2 - 3xL + 3x^2\right)\right] - \left(vm^3L/3n\right)\left[F_p(L-x)^3\right]\left[1 - x/L + px/L\right]$$

$$B_2 = -\left(1/6L^2\right)\left[4F_p(L-x)(L-2x)Lx\right] - \left(m^3L\,F_p/3n\right)(2x-L) - (p-1)vm^3F_p(L-x)^3/3n$$

where: n = i2/I1, m=a1/L, I1, I2 - moments of inertia of shaft section and jaw, respectively; n = ∞ - it is assumed that the jaws do not undergo deformation; p = Δl/Δp - relative displacement of ends of the part due to inaccuracy of geometric alignment of semi-finished product, p = Δ*l/Δ*p - relative displacement of end of the part as a result of jaw deflection from machining force Fp; v = p = 0 F_p - inaccuracy of initial alignment is not taken into account [2,22].

Deformations from all kinds of loading of jaws - left - Δl and right - Δp - are determined. At n = ∞, v = p ≠ 0 and M1 = 0 there occurs a specific case of control of only the right bending moment, and therefore the deflection of the part at the point of contact with the tool should equal zero: $M_2(x) = 2F_p x(x-L)/(x+L)$.

At n = ∞ , $v = p = 0$ and $\Delta^* = 0$ (Δ^*- displacement resulting from deflection), M ≠ 0, there occurs a specific case of control with absolutely rigid jaws (curves 1, Fig. 3b); $n = 1$, $v = p = 0$, $m = 1/4$ - curves 2; $v = p = 1$ - curves 3. In the latter two cases, bending moments at $x = 0$ and $x = L$ are different from zero, and their maximum absolute values increase due to the necessity of compensation of displacement of the section of the semi-finished product resulting from deflection of the jaws.

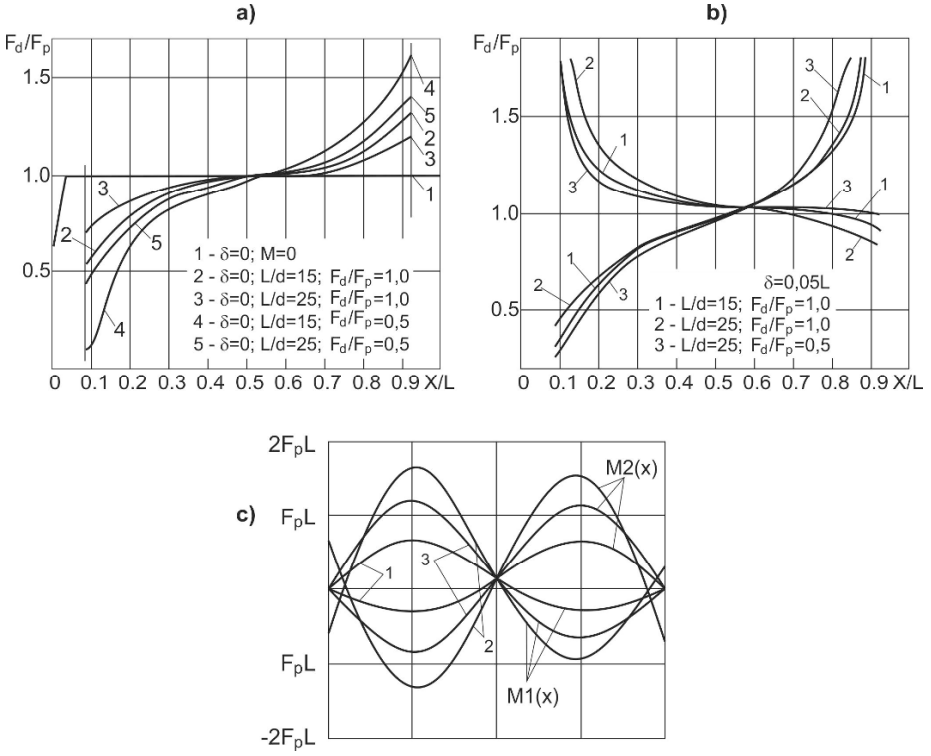


Fig. 2. Graphs of relation F_d/F_p *(x/L)* in the case of use of lever-type vibration dampers at: a) δ = 0, $M_1 = 0$; b) $M_1 \neq 0$, δ = 0,05 L , c) graphs of bending moments *M(x)* in the process of grinding

With condition (1) fulfilled, we obtained the relations of change of F_d in the case of various sections at $F_{d1} = F_{d2} = F_{d3} = F_{d4} = F_d$, $M_1 = F_f d/2$, section 0≤x≤(L-3B):

$$F_d(x) = \frac{2F_p x^3\left(x^3 - 9xL^2 + 8L^3\right) + 3M_1 x^2\left(6xL^2 - 3x^2 L - 2L^3\right)}{\left(18x^2L^2 - 6x^3L\right)\left(L^2 + 5b^2\right) + \left(x^3 - 3x^2L\right)\left(L^3 + 6Lb^2\right) - 6x^2L^4 + 12x^3L^3} \tag{9}$$

section (L-3b)/≤ v ≤(L-b)/2

$$F_d(x) = \frac{2F_p x^3\left(x^3 - 9xL^2 + 8L^3\right) + 3M_1 x^2\left(6xL^2 - 3x^2 L - 2L^3\right)}{\left(18x^2L^2 - 6x^3L\right)\left(L^2 + 5b^2\right) + \left(x^3 - 3x^2L\right)\left(L^3 + 6Lb^2\right) - 6x^2L^4 + 12x^3L^3 - 4L^3\left(x - \frac{L-3b}{2}\right)^3} \tag{10}$$

section *(L-b)/2 ≤ x ≤ (L+b)/2*.

$$F_d(x) = \frac{2F_p x^3\left(x^3 - 9xL^2 + 8L^3\right) + 3M_1 x^2\left(6xL^2 - 3x^2 L - 2L^3\right)}{\left(18x^2L^2 - 6x^3L\right)\left(L^2 + 5b^2\right) + \left(x^3 - 3x^2L\right)\left(L^3 + 6Lb^2\right) - 6x^2L^4 + 12x^3L^3 - 4L^3\left[\left(x - \frac{L-3b}{2}\right)^3 + \left(x - \frac{L-b}{2}\right)^3\right]} \tag{11}$$

section *(L +b)/2 ≤ x ≤ (L + 3b)/2*

$$F_d(x) = \frac{2F_p x^3\left(x^3 - 9xL^2 + 8L^3\right) + 3M_1 x^2\left(6xL^2 - 3x^2 L - 2L^3\right)}{\left(18x^2L^2 - 6x^3L\right)\left(L^2 + 5b^2\right) + \left(x^3 - 3x^2L\right)\left(L^3 + 6Lb^2\right) - 6x^2L^4 + 12x^3L^3 - 4L^3 A} \tag{12}$$

$$A = \left[\left(x - \frac{L-3b}{2}\right)^2 + \left(x - \frac{L-b}{2}\right)^3 + \left(x - \frac{L+b}{2}\right)^3\right] \tag{13}$$

section *(L +3b)/2 ≤ x ≤ L*

$$F_d(x) = \frac{2F_p x^3\left(x^3 - 9xL^2 + 8L^3\right) + 3M_1 x^2\left(6xL^2 - 3x^2 L - 2L^3\right)}{\left(18x^2L^2 - 6x^3L\right)\left(L^2 + 5b^2\right) + \left(x^3 - 3x^2L\right)\left(L^3 + 6Lb^2\right) - 6x^2L^4 + 12x^3L^3 - 4L^3 B} \tag{14}$$

$$B = \left[\left(x - \frac{L-3b}{2}\right)^3 + \left(x - \frac{L-b}{2}\right)^3 + \left(x - \frac{L+b}{2}\right)^3 + \left(x - \frac{L+3b}{2}\right)^3\right] \tag{15}$$

Applying the theorem of three moments, in the case of the first and the second stop the equation can be written in the form:

$$\left.\begin{array}{c} 4M_1 + M_2 L = F_p a\left(L^2 - x^2\right)/L \\ M_1 L + 4M_2 L = 0 \end{array}\right\} \tag{16}$$

where: M_1 and M_2 – stop bending moments.

From relation (16) we obtained the values of M_1, M_2, and the reaction in stop A.

From the differential equation of bent axis $y'' = \frac{1}{EI}\left[(-R_A x) + F_p(x-a)\right]$ we obtained the relations for the determination of deflections of the shaft an any point and at the point of application of force x = a. Analysis of the relations obtained shows that unsupported section of the shaft increase the greatest deflections in the machining zone by 1/3 compared to twin-stop shafts.

3 Conclusions

The paper presents an effective method of improving the accuracy of forming of elastic-deformable shafts with low rigidity on the basis of developed relations allowing the regulation of elasticity and the control of the technological system in the process of shaft machining.

The technological method of control of forming accuracy, based on regulation of the elastic-deformable state of parts with low rigidity as a result of the action, in the conditions of cross-bending, of additional force factors oriented at the compensation of forces generated in the process of machining, is presented and substantiated.

The relations that allow the determination of additional force at which it is possible to achieve uniform rigidity on the length of the shaft, at the point of application of machining force, and relations for the control of bending moments, allowing the achievement of minimum bending deformations in the machining zone, taking into account the stiffness of the stops and the geometric inaccuracy of alignment, were determined.

Vibration dampers are used to increase the accuracy of machining of shafts. Analysis showed that moving the vibration damper to the left or to the right from the point of application of machining force causes the necessity of increasing the additional force required to compensate the bending deformations from the machining force. With the application of twin-stop vibration dampers the quotient of the additional force and the thrust force of machining decreases approximately two-fold compared to single-stop dampers.

The presented relations allow minimization as well as stabilization of the value of elastic deformations of shafts with low rigidity on their entire length, which allows enhancement of the accuracy and efficiency of production of elastic-deformable shafts with low rigidity, and thus also a significant improvement of their operational properties.

References

1. Świć, A., Wołos, D., Litak, G.: Method of control of machining accuracy of low-rigidity elastic-deformable shafts. Latin American Journal of Solids and Structures. 11(2), 260-278 (2014)
2. Swic, A., Taranenko. W.: Adaptive control of machining accuracy of axial-symmetrical low-rigidity parts in elastic-deformable state. Eksploatacja i Niezawodnosc – Maintenance and Reliability. 14 (3), 215-221 (2012)

3. Tlusty, J.: Manufacturing processes and equipment. Upper Sadde River, NJ: Prentice Hall, 2000.
4. Bajić, D., Celent, L., Jozić, S.: Modeling of the influence of cutting parameters on the surface roughness, tool wear and cutting force in face milling in off-line process control. Stroj Vestn-J Mech E. 58, 673-682 (2012)
5. Urbicain, G., Olvera, D., Fernández, A., Rodríguez, A., Tabernero, I., López de Lacalle, L.N.: Stability Lobes in Turning of Low Rigidity Components. Advanced Materials Research. 498, 231-236 (2012)
6. Świć, A., Draczew, A., Gola A.: Method of achieving accuracy of thermo-mechanical treatment of low-rigidity shafts. Advances in Science and Technology-Research Journal, 10 (29), 62-70 (2016)
7. Campa, F. J., de Lacalle, L.N.L., Urbikain, G., Ruiz, D.: Definition of cutting conditions for thin-to-thin milling of aerospace low rigidity parts. Proceedings of the ASME International Manufacturing Science and Engineering Conference 2008. 1, 359-368 (2008)
8. Ryu, S. H., Lee, H. S., Chu, C. N.: The form error prediction in side wall machining considering tool deflection. Int. J Mach Tools Manuf. 43, 731-737 (2003)
9. Ratchev, S., Liu, S., Huang, W. Becker, A.A.: Milling error prediction and compensation in machining of low-rigidity parts. Int J Mach Tools Manuf. 44 (15), 1629-1641 (2004)
10. Tian, L.Z., Wu J.H., Xiong Z.H., Ding H.: Active chatter suppression in turning of low-rigidity workpiece by system matching. Lect Notes in Art Intel. 9245, 609-618 (2015)
11. Qi, H., Tian, Y., Zhang, D.: Machining forces prediction for peripheral milling of low-rigidity component with curved geometry. Int J Adv Manuf Technol. 64, 1599-1610 (2013)
12. Campomanes, M. L., Altintas, Y.: An improved time domain simulation for dynamic milling at small radial immersions. Trans ASME, J Manuf Sci Eng. 125, 416-422 (2003)
13. Li, H., Shin, Y.C.: A comprehensive dynamic and milling simulation model. Trans ASME, J Manuf Sci Eng.128, 86-95 (2006)
14. Lorong, P., Coffignal, G., Cohen-Assouline, S.: Simulation du comportement dynamique d'un systeme usinant: modelisation de l'interaction outil/matiere en presence d'une piece flexible. Mec Ind. 9, 117-124 (2008)
15. Altintas, Y: Manufacturing automation: metal cutting mechanics, machine tool vibrations, and CNC design. Cambridge: Cambridge University Press (2000)
16. Chen C.K., Tsao Y.M.: A stability analysis of regenerative chatter in turning process without using tailstock. Int J Adv Manuf Technol. 29, 648–654 (2006)
17. Hassui, A., Diniz, A.E.: Correlating surface roughness and vibration on plunge cylindrical grinding of steel. Int J Mach Tools Manufac. 43, 855–862 (2003)
18. Świć, A., Wołos, D., Zubrzycki, J., Opielak, M., Gola, A., Taranenko, V.: Accuracy Control in the Machining of Low Rigidity Shafts. Applied Mechanics and Materials. 613, 357-367 (2014)
19. Litak, G., Rusinek, R., Teter, A.: Nonlinear analysis of experimental time series of a straight turning process. Meccanica. 39, 105–112 (2004)
20. Świć, A., Gola, A., Wołos, D., Opielak, M.: Micro-geometry Surface Modelling in the Process of Low-Rigidity Elastic-Deformable Shafts Turning. Iranian Journal of Science and Technology, Transactions of Mechanical Engineering. 1-9 (2016)
21. Cardi, A.A., Firpi, H.A., Bement, M.T., Liang, S.Y.: Workpiece dynamic analysis and prediction during chatter of turning process. Mech Syst Signal Pr. 22, 1481–1494 (2008)
22. Qiang, L.Z.: Finite difference calculations of the deformations of multi-diameter workpieces during turning. J Mat Process Technol. 98, 310–316 (2000)
23. Jianliang, G, Rongdi, H.: A united model of diametral error in slender bar turning with a follower rest. Int J Mach Tools Manufac. 46, 1002–1012 (2006)

A Negotiation Algorithm for Decision-Making in the Construction Domain

Arazi Idrus[1], Moamin A. Mahmoud[2], Mohd Sharifuddin Ahmad[2], Azani Yahya[1], Hapsa Husen[1]

[1] Department of Civil Engineering, Universiti Pertahanan Nasional Malaysia
Kem Sungai Besi, 57000 Kuala Lumpur, Malaysia
[2] College of Computer Science and Information Technology,
Universiti Tenaga Nasional, Kajang, Selangor, Malaysia.

arazi@upnm.edu.my,{moamin, sharif }@uniten.edu.my, {azani, hapsa}@upnm.edu.my

Abstract. In this paper, we propose a negotiation algorithm for automated decision-making in the construction domain. The algorithm enables software agents to conduct negotiations and autonomously make decisions. The algorithm consists of eight steps which are submit solutions, review solutions' weights, solutions pair ranking, set a plan, select the plan, form coalition, review coalitions' strength, and re-evaluate coalition. We first present the algorithm architecture then the related definition and formulation of each step and subsequently demonstrate the complete operation of the proposed algorithm using pseudocode.

Keywords: Multi-agent Systems, Automated Negotiation, Construction Domain.

1 Introduction

In the construction domain [1, 2], a project manager usually cares more about the cost of a project than the function while a design manager is more concerned about the function than the cost. Thus, for any decision to be made regarding a new project, stakeholders must propose a single optimal solution. However, a problem may arise when stakeholders propose more than one solution. In such situation, stakeholders need to negotiate on the proposed solutions and agree on a single solution. But the negotiation may not be easy and smooth because when stakeholders possess different backgrounds, often their views about the optimal solution for a particular project are different. Such differences cause conflicts in arriving at a decision. In addition, stakeholders may work at different branches throughout the country or other parts of the world which make a meeting for decision more difficult and costly.

In this paper, we attempt to overcome these difficulties by proposing a negotiation algorithm for automated decision-making in the construction domain. The proposed

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_14

algorithm enables software agents to conduct negotiations and autonomously make decisions. The algorithm consists of eight steps which are submit solutions, review solutions' weights, solutions pair ranking, set a plan, select the plan, form coalition, review coalitions' strength, and re-evaluate coalition. This paper is an extension to our work in automated multi-agent negotiation [3, 4, 5].

2 Related Work

Intelligent software agents have been widely used in distributed artificial intelligence and due to their autonomous, self-interested, rational abilities [6, 7, 8, 9, 10, 11], and social abilities [12, 13, 14, 15], agents are well-suited for automated negotiation on behalf of humans [16]. According to Kexing [16], automated negotiation is a system that applies artificial intelligence and information and communication technology to negotiation strategies, utilizing agent and decision theories.

Numerous research have discussed the works in negotiation on multi-agent systems in various domains [17, 18, 19, 20,21]. Coutinho et al. [22] proposed a negotiation framework to serve collaboration in enterprise networks to improve the sustainability of interoperability within enterprise information systems. Utomo [1, 2] presented a conceptual model of automated negotiation that consists of methodology of negotiation and agent-based negotiation. Dzeng and Lin [23] presented an agent-based system to support negotiation between construction and suppliers via the Internet. Anumba et al. [24] proposed a collaborative design of light industrial buildings based on multi-agent systems to automate the interaction and negotiation between the design members. Ren et al. [18] developed a multi-agent system representing participants, who negotiate with each other to resolve construction claims.

3 Negotiation algorithm (A_{NEGO})

The algorithm implements a negotiation process between agents. The process starts when each agent proposes ranked solutions and submits them to a negotiation base. Proposing ranked solutions by agents is discussed in [25]. Each agent then reviews the submitted solutions and accordingly sets a plan to conduct the negotiation. The A_{NEGO} algorithm consists of eight steps as shown in Figure 1. We detail out these steps as follows:

Step 1: Submit Solutions - After each agent generates and ranks its solutions, it submits them to the Negotiation Base. As mentioned earlier, the agent submits its ranked solutions in the form,
agentName (solutionNumber, solutionRank).

Step 2: Review Solutions' Weights - In this step, agents review each solution's weight from the Negotiation Base.

Definition 1: A Solution Weight, L_W, represents the adoption factor of a solution by agents in different ranks. In other words, the weight for a particular solution is based on how many agents adopt that solution in each rank, e.g. a solution L is adopted by four agents in rank 1, three agents in rank 2, …, and seven agents in rank n.

Definition 2: A Coalition Strength, ST_{Cn}, is the percentage number of agents that have allied to form a coalition based on the total number of agents.

Definition 3: Negotiation Styles and Outcomes, there are five main negotiation styles that constitute two types of outcomes, which are Competing, Avoiding, Collaborating, Accommodating, and Compromising, and the two outcomes are Cooperative and Assertiveness. Accordingly, each agent, based on its style, has a negotiation outcome which is either Cooperative or Assertive. Each style maintains each outcome at one of three levels (High, Medium, or Low). For example, the Competing style maintains Low Cooperative outcome but High in Assertiveness. Consequently, an agent's negotiation outcome is based on its style. When an agent is High in Assertiveness, then the agent often possesses the Competing style because its assertiveness dominates its cooperativeness. Conclusively, each agent possess one style, and this style forms one negotiation outcome whether it is Cooperative or Assertiveness, e.g. agent α is Cooperative type and possess Accommodating style.

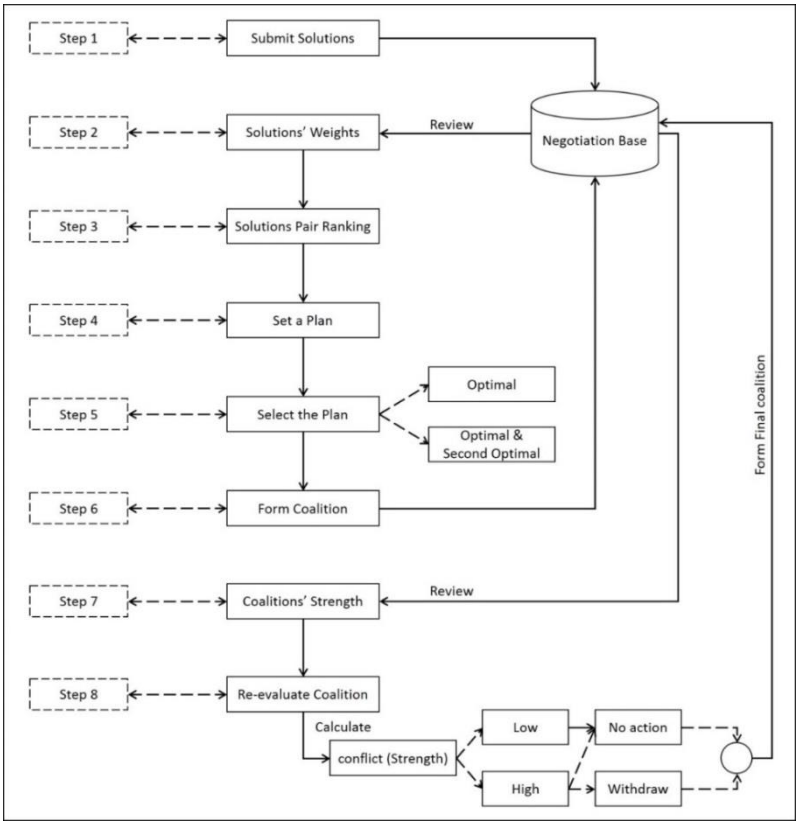


Figure 1. The Negotiation Algorithm (ANEGO) Process

Calculation of Weight: The Negotiation Base calculates the weight of each solution and ranks them from highest weight 1st to the lowest weight nth in the form, solutionNumber(solutionWeight, rank).

Each agent is able to review the solutions' weights and ranks from the Negotiation Base as shown in Figure 2. From this review, an agent sets a plan.

The weight for a particular solution is based on the number of agents that adopt a solution in each rank. If A_D is the total number of agents that adopt a same solution in a same rank, A_T is the total number of agents that are involved in the negotiation, then the percentage of agents that adopt a solution in a same rank = A_D/A_T. Since agents normally attempts strongly to get their first rank solution win, then logically, the weight of the first rank solution is greater than the weight of the second rank solution and so on. Therefore, each order is multiplied by $1/2^R$ where R is the rank number = 1, 2, 3, …, so that,

Solution Weight, $$L_{nW} = \frac{\left(\left(\frac{1}{2^R}\right)* A_D(L_1R)\right)+\left(\left(\frac{1}{2^R}\right)* A_D(L_2R)\right)+ \ldots.+\left(\left(\frac{1}{2^R}\right)* A_D(L_nR)\right)}{A_t}$$

$$L_{nW} = \frac{\sum_{R=1}^{\infty}\left(\frac{1}{2^R}\right)* A_D(L_nR)}{A_T}$$

where $A_D(L_nR)$ is the number of agents that adopt the solution n in rank R and n is any natural number.

Step 3: Solutions Pair Ranking - Each agent currently has two rankings for each solution, the first one is based on its desire, and the second is based on the solution weight which is calculated by the Negotiation Base. An agent outlines the ranks of each solution in the form,
solutionNRank (solutionDesire, solutionWeight)

e.g. solution1Rank(2, 3), where 2 is the solution rank based on the agent desire (the second choice), and 3 is the solution rank based on the solution weight (third highest weight).

Step 4: Set a Plan - An agent sets a plan based on the solutionNRank pair of each solution and its type whether it is cooperative or assertive agent. The agent first finds the mean of each solutionNRank, if L_nR_D is solutionDesire rank, L_nR_W is solutionWeight rank, μ_{Sn} is the mean of the solution n, then

$$\mu_{Sn} = \frac{L_nR_D + L_nR_W}{2} \qquad 1$$

From the above example, for solution1Rank(2, 3), the mean, $\mu_{S1} = \frac{2+3}{2} = 2.5$

Step 5: Select the Plan - From the means of all solutions, the agent selects its plan. The lowest mean value is considered as the optimal solution. Thus, the mean value of 1 is considered the lowest possible mean for a solution because it comes from solutionNRank(1,1), where solutionDesire rank equals 1 (first choice), and solutionWeight rank equals 1 (first highest weight), and then the mean equals μ_{Sn} =

$\frac{1+1}{2} = 1$. Consequently, the lowest mean value is considered a better solution than a higher one. However, if there are two means that hold the same lowest value, e.g. solution1Rank(2, 3), solution2Rank(3, 2), then the mean value equals 2.5 for both. In this case,

- If the agent is a cooperative type, it then sets the solution which holds higher rank of solutionDesire (solution1Rank in the example) as an optimal solution and the other solution which holds lower rank of solutionDesire (solution2Rank in the example) as a second optimal solution.
- If the agent is an assertive type, it only sets the optimal solution which holds higher rank of solutionDesire (solution1Rank in the example).

Ultimately, each agent has a plan. The plan is either the optimal solution only, or the optimal and second optimal solutions. If P is a plan then, S_{OP} is the optimal solution, S_{OP2} is the second optimal solution, then,

$$P = [\, S_{OP}(S_{OP}, S_{OP2})]$$

Step 6: Form Coalition, C - When the plan of an agent is ready, it forms coalition by adding +1 to its optimal solution in the Negotiation Base.
Step 7: Review Coalitions' Strength - The Negotiation Base calculates the strength of each coalition and ranks them from highest strength 1st to the lowest strength nth in the form,
coalitionName(coalitionStrength, rank)

Each agent is able to review the coalitions' strengths and ranks from the Negotiation Base as shown in Figure 2. From this review, an agent re-evaluates its coalition.

Calculation of Coalition Strength: The strength equals the number of agents of a specific coalition, C_A, divided by the total number of agents, A_T,

$$ST_{Cn} = C_A/A_T \qquad 2$$

Step 8: Re-evaluate Coalition - Each agent re-evaluates its coalition according to other coalition's strengths.

From ST_{Cn} value,

- If the agent's coalition strength rank is 1st or 2nd, then no action is necessary because the next stage (conflict resolution) processes the 1st or 2nd with more similar opportunity for acceptance.
- If the rank is 3rd onward, it then calculates the conflict strength, ST_{CNF}, with the higher coalitions strengths, e.g. if the coalition strength rank is 3rd, then the agent calculates the conflict strength between 1st (ST_{C1}), 2nd (ST_{C2}), and 3rd (ST_{C3}). We exploit the Standard Deviation STDV function to calculate the conflict strength, ST_{CNF}:

$ST_{CNF} = STDV(ST_{C1}, ST_{C2}, .., ST_{Cn})$ 3

The result of ST_{CNF} is either high conflict or low conflict depends on a value called the conflict strength threshold value, T_{STV}. However, the T_{STV} value is indeterminate at this juncture, therefore in the next sections, we denote it as T_{STV}.

- If the result is $ST_{CNF} \leq T_{STV}$ (low conflict), then no action by the agent is necessary because its coalition strength is marginally close to other coalitions' strength, which means that the coalitions have similar opportunity to be accepted.
- If the result is $ST_{CNF} > T_{STV}$ (high conflict), then the agent would withdraw from its coalition and join other coalitions with higher strengths (e.g. 1st or 2nd), which adopts a solution that suits its solutions rank. An agent's decision to remain or withdraw is based on its characteristic, i.e. cooperative or assertive. If it is a cooperative type, the agent withdraws and processes the second optimal solution, and remains otherwise. However, agents' characteristics determination is beyond the scope of this research.

Ultimately, agents have formed coalitions and each coalition presents a single solution. Algorithm 2 shows the process of the Negotiation Algorithm (A_{NEGO})

Algorithm 1: The Negotiation Algorithm (A_{NEGO})

```
1.  begin
2.    for all agents negotiate solutions Do
3.    employ (agents, A_NEGO)
4.    step 1: submit(agents, solutions)
5.      agentName (solutionNumber, solutionRank) → N_BASE
6.    step 2: review (agent, solutionsWeight) ← N_BASE,
7.      calculate (N_BASE, L_nw)
8.      L_nW = (Σ_{R=1}^{∞} (1/2^R) * A_D(L_nR)) / A_T,
9.    step 3: rank(agent, solutions) based on (desire, weight)→
10.     solutionRank (solutionDesire, solutionWeight)
11.   step 4: set(agent, plan)→
12.     for each solution, find(agent, μ_Sn) →
13.     μ_Sn = (L_n R_D + L_n R_W) / 2
14.   step 5: select (agent, plan)
15.     if (one μ_Sn holds lowest value)
16.       set(S_OP),
17.     else
18.       if (agentCooperative)
19.         set (S_OP, S_OP2),
20.       else
21.         set (S_OP)
22.       end_if
23.     end_if
```

```
24.    step 6: form(agent, coalition)
25.       AgentName (+1, coalitionSoliution)
26.    step 7: review (agent, coalitionStrenght) ← N_BASE,
27.       calculate (N_BASE, ST_Cn)
28.       ST_Cn = C_A/A_T,
29.    step 8: re-evaluate(agent, coalition)
30.       if (ST_Cn rank = 1st || ST_Cn = 2nd )
31.          no action
32.       else
33.          calculate(agent, ST_CNF)
34.          ST_CNF = STDV(C_1, C_2, .., C_n).
35.          determine (agent, T_STV),
36.          if (ST_CNF ≤ T_STV )
37.             no action
38.          else
39.             if (agentCooperative)
40.                withdraw & join another,
41.             else agentAssertive
42.                no action,
43.             end if
44.          end if
45.       end if
46. end
```

4 Conclusion and Further Work

In this paper, we present the findings of our work-in-progress to develop an automated multi-agent negotiation model for decision making in the construction domain. We propose a negotiation algorithm that enables software agents to conduct negotiations and autonomously make decisions. The algorithm consists of eight steps which are submit solutions, review solutions' weights, solutions pair ranking, set a plan, select the plan, form coalition, review coalitions' strength, and re-evaluate coalition. We present the algorithm and the related definitions and formulation of each step. We have also demonstrated the complete operation of the proposed algorithm using a pseudocode.

Since this work is in its theoretical stage, it only presents the algorithmic approach in negotiation and does not present the experimental results. Such outcome will be presented in our future work.

5 References

1. Utomo C., Development of a negotiation Support Model for Value Management in Construction, PhD Thesis, University Teknologi PETRONAS, December 2009.

2. Utomo C., Idrus A., A Concept toward Negotiation Support for Value Management on Sustainable Construction, Journal of Sustainable Development Vol 4, No 6 (2011).
3. Mahmoud, M. A., Ahmad, M. S., Yuso, M. Z. M., & Idrus, A. (2015). An Automated Negotiation-based Framework via Multi-Agent System for the Construction Domain. International Journal of Artificial Intelligence and Interactive Multimedia, 3(5), 23-27.
4. Mahmoud, M. A., Ahmad, M. S., Yusoff, M. Z. M., & Idrus, A. (2015). Automated Multi-agent Negotiation Framework for the Construction Domain. In Distributed Computing and Artificial Intelligence, 12th International Conference (pp. 203-210). Springer International Publishing.
5. Mahmoud, M. A., Ahmad, M. S., & Yusoff, M. Z. M. "A Conceptual Automated Negotiation Model for Decision Making in the Construction Domain." Distributed Computing and Artificial Intelligence, 13th International Conference. Springer International Publishing, 2016.
6. Ahmed M., Ahmad M S, Yusoff M Z M, Modeling Agent-based Collaborative Process , The 2nd International Conference on Computational Collective Intelligence Technology and Applications (ICCCI 2010), pp. 296-305, ISBN:3-642-16692-X 978-3-642-16692-1, 10-12 November, 2010 Taiwan.
7. Mahmoud, M. A., & Ahmad, M. S. (2015, August). A self-adaptive customer-oriented framework for intelligent strategic marketing: A multi-agent system approach to website development for learning institutions. In Agents, Multi-Agent Systems and Robotics (ISAMSR), 2015 International Symposium on (pp. 1-5). IEEE.
8. Itaiwi A. K., Ahmad M. S., Hamid N. H. A., Jaafar N. H., Mahmoud M. A., A Framework for Resolving Task Overload Problems Using Intelligent Software Agents, 2011 IEEE International Conference on Control System, Computing and Engineering, ICCSCE11,2011.
9. Ahmed M., Ahmad M. S., and Yusoff M. Z. M., "A Collaborative Framework for Multiagent Systems." International Journal of Agent Technologies and Systems (IJATS), 3(4):1-18, 2011.
10. Ahmed M., Ahmad M. S., and Yusoff M. Z. M.,, Mitigating Human-Human Collaboration Problems using Software Agents, The 4th International KES Symposium on Agents and Multi-Agent Systems – Technologies and Application (AMSTA 2010), pp. 203-212, ISBN:3-642-13479-3 978-3-642-13479-1, Gdynia, Poland, 23 – 25 June 2010.
11. Jassim, O. A., Mahmoud, M. A., & Ahmad, M. S. (2015). A Multi-agent Framework for Research Supervision Management. In Distributed Computing and Artificial Intelligence, 12th International Conference (pp. 129-136). Springer International Publishing.
12. Mahmoud, M. A., Ahmad, M. S., Mohd Yusoff, M. Z., & Mustapha, A. (2014). A Review of Norms and Normative Multiagent Systems. The Scientific World Journal, 2014.
13. Mahmoud, M. A., Ahmad, M. S., & Yusoff, M. Z. M. "Development and implementation of a technique for norms-adaptable agents in open multi-agent communities." Journal of Systems Science and Complexity 29.6 (2016): 1519-1537.
14. Mahmoud, M. A., Ahmad, M. S., & Yusoff, M. Z. M. (2016, March). A Norm Assimilation Approach for Multi-agent Systems in Heterogeneous Communities. In Asian Conference on Intelligent Information and Database Systems (pp. 354-363). Springer Berlin Heidelberg.
15. Mahmoud, M. A., Mustapha, A., Ahmad, M. S., Ahmad, A., Yusoff, M. Z. M., & Hamid, N. H. A. (2013). Potential norms detection in social agent societies. In Distributed Computing and Artificial Intelligence (pp. 419-428). Springer International Publishing.

16. Kexing L.. A survey of agent based automated negotiation. In Network Computing and Information Security (NCIS), 2011 International Conference on, vol. 2, pp. 24–27 (IEEE, 2011).
17. Beer M., d'Inverno M., Jennings R.N., Luck M., Preist C., Schroeder M., Negotiation in multi-agent systems Knowledge Engineering Review, 14 (3) (1999), pp. 285–289
18. Z. Ren, C.J. Anumba, Multi-agent systems in construction—state of the art and prospects, Automation in Construction, 13 (2004), pp. 421–434
19. M. Wang, H. Wang, D. Vogel, K. Kumar, D.K.W. Chiu Agent-based negotiation and decision making for dynamic supply chain formation Engineering Applications of Artificial Intelligence, 22 (7) (2009), pp. 1046–1055
20. Utomo C., Idrus A., A Concept toward Negotiation Support for Value Management on Sustainable Construction, Journal of Sustainable Development Vol 4, No 6 (2011).
21. Victor Sanchez-Anguix , Vicente Julian , Vicente Botti , Ana García-Fornes, Tasks for agent-based negotiation teams: Analysis, review, and challenges, Engineering Applications of Artificial Intelligence, v.26 n.10, p.2480-2494, November, 2013.
22. Coutinho, C., Cretant, A., Ferreira da Silva, C., Ghodous, P., & Jardim-Goncalves, R. (2014). Service-based negotiation for advanced collaboration in enterprise networks. Journal of Intelligent Manufacturing. doi:10.1007/s10845-013-0857-4.
23. Dzeng, R. J., & Lin, Y. C. (2004). Intelligent agents for supporting construction procurement negotiation. Expert Systems with Applications, 27(1), 107-119.
24. C.J. Anumba, Z. Ren, A. Thorpe, O.O. Ugwu, L. Newnham, Negotiation within a multi-agent system for the collaborative design of light industrial buildings Adv Eng Software, 34 (7) (2003), pp. 389–401
25. Idrus, A, Mahmoud, M. A., Ahmad, M. S., Yahya A, Husen H., A Solution Generator Algorithm for Decision Making based Automated Negotiation in the Construction Domain, International Journal of Artificial Intelligence and Interactive Multimedia, 2017.

Deep neural networks and transfer learning applied to multimedia web mining

Daniel López-Sánchez, Angélica González Arrieta, Juan M. Corchado

Department of Computer Science and Automation, University of Salamanca, Spain

Abstract. The growth in the amount of multimedia content available online supposes a challenge for search and recommender systems. This information in the form of visual elements is of great value to a variety of web mining tasks; however, the mining of these resources is a difficult task due to the complexity and variability of the images. In this paper, we propose applying a deep learning model to the problem of web categorization. In addition, we make use of a technique known as transfer or inductive learning to drastically reduce the computational cost of the training phase. Finally, we report experimental results on the effectiveness of the proposed method using different classification methods and features from various depths of the deep model.

Keywords: Web mining, deep learning, transfer learning

1 Introduction

During the last decades, the number of web pages available on the internet has grown exponentially. The proliferation of blog-hosting and free content management systems (CMS) such as WordPress, Blogger or Tumblr have contributed to this growth by making it possible for users with no experience in managing digital systems to share a variety of contents. The nature of these hosting services promotes the publishing of multimedia contents, especially images and videos. However, this democratization of the internet has originated new and challenging problems. Specifically, it has become increasingly difficult for users to find the content that they demand while avoiding related but undesired results. In addition, the availability of abundant multimedia content supposes challenge to recommender and search systems. Conventional recommender and search systems do not usually take into account the discriminative information provided by multimedia content, limiting themselves to the analysis of textual information. This is because mining multimedia data is a highly complex problem that frequently demands intensive computations and large training datasets.

On the other hand, the field of artificial vision, and specifically the sub-field of visual object recognition, has experienced a major breakthrough after the general adoption of the deep learning paradigm in recent years [10]. Deep learning models are composed of a previously intractable number of processing layers, which allows the models to learn more complex representations of the data by taking into consideration multiple levels of abstraction. This eventually led to a dramatic improvement in the state-of-the-art of visual object recognition, object detection and other related domains [4]. The keys

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_15

to success for deep learning are the complexity of the models and the availability of large datasets with training data. The major drawback of deep learning techniques is their computational cost both in training and test phases.

In this paper, we propose applying a state-of-the-art model in visual object recognition to the field of multimedia web mining. Specifically, we propose using a deep convolutional neural network (DCNN) to the task of web page categorization based on the available multimedia content. To overcome the problems of computational cost and the need for large training datasets, we propose using a technique called transfer learning, which makes it possible to use the knowledge gained while solving one problem to another different but somehow related problem.

2 Proposed method

In this section the proposed method is described in detail. First, the global structure and elements of the processing pipeline are presented, followed by a detailed description of each element in the chain. Our system is designed to perform the following task: given an URL, the system must (1) access that URL, extract all the images available in the web page, and filter those that do not contain discriminative information; (2) extract a feature descriptor from each image such that the classification problem becomes easier over that feature space; and (3) analyze each feature descriptor and combine the results to emit a prediction concerning the category of the web page. This process is schematically shown in Figure 1.

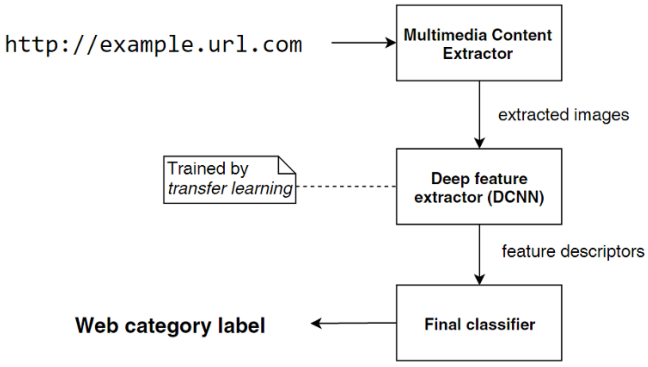


Fig. 1. Steps of the proposed method

2.1 Multimedia content extractor

The first module of the proposed method is in charge of extracting all the images that are present in a given web page and filter the ones that do not contain useful information (i.e. advertisements, banners, etc.). To do this, the system begins by downloading the HTML document to which the provided URL points. Then, we use the Beautiful Soup library [7] to analyze the structure and the hierarchy in the document. After this, the web page is represented as a tree whose leaves are the elements of the document (e.g.

titles, links, images, etc.). This representation of the document is explored exhaustively in order to find image elements; the URLs that point to those images are stored and later used to download the pictures.Several criteria can be applied to filter the extracted images; for example, it is possible to discard the images that contain a specific group of keywords in the alt attribute (e.g. we might discard images that contain the word "advertisement" in its alt attribute). Another possible approach consists of rejecting the images whose dimensions are outside a specific range. This is because images with rare proportions do not usually contain discriminative information. For example, very small images might correspond to navigation icons, and images of elongated shape tend to be advertisement banners.

2.2 Deep feature extractor

Once a number of images have been extracted from a web page, it would be possible to directly apply any classification method. However, the complexity of the image recognition problem that we are trying to solve demands a large number of training instances and a very complex model that can manage the difficulty of the classification problem. This is mainly due to the high intra-class variability of the samples from such artificial vision problems. Collecting such an extensive dataset can be a tedious and very time-consuming task. In addition, the training time of such a classifier would be very long even if we used complex computation parallelization techniques and expensive devices.

To overcome these limitations, we propose applying a technique known as transfer learning (see [8] for a recent survey on the topic). The key idea of this method is to apply the knowledge gained while solving one problem to a different but related problem. In practice, this technique has been mainly applied in the context of artificial neural networks. Here, some of the layers of the network are initialized with the weights learned by another network that was trained to solve a different problem; the remaining weights are initialized at random (as usual). A short training phase is then executed to tune the weights of the network that were randomly initialized. While the transferred layers may also be fine-tuned by executing various iterations of the chosen training algorithm, in principle this step is not mandatory. Avoiding this adjustment process will allow us to use any kind of classifier on top of the transferred neural layers, even if it is not compatible with the backpropagation method.

In the case study for this paper, we propose using a pre-trained model of DCNN. Specifically, the selected model is the VGG-16 DCNN [11] developed by the Visual Geometry Group at the University of Oxford[1]. The VGG-16 network achieved second place in the classification task of the Large Scale Visual Recognition Challenge 2014[2].

---

[1] The VGG16 model is available at the Visual Geometry Group web page: http://www.robots.ox.ac.uk/~vgg/research/very_deep/

[2] Large Scale Visual Recognition Challenge 2014 web page: http://www.image-net.org/challenges/LSVRC/2014/

It was trained on the ImageNet dataset, which consists of 14 million images belonging to 1000 categories; the model achieves 92.7% top-5 test accuracy on this benchmark. Although several models have outperformed VGG16, this model remains competitive with the state-of-the-art. In addition, it was designed with computational costs in mind. The number of parameters of the network was significantly reduced by using small 3×3 kernels in the convolution layers of the network. We chose the VGG16 model because it maintains a balance between computational costs and accuracy. Figure 2 shows the overall architecture of the network; we will refer to this figure to name each of the layers of the model in the experimental results section.

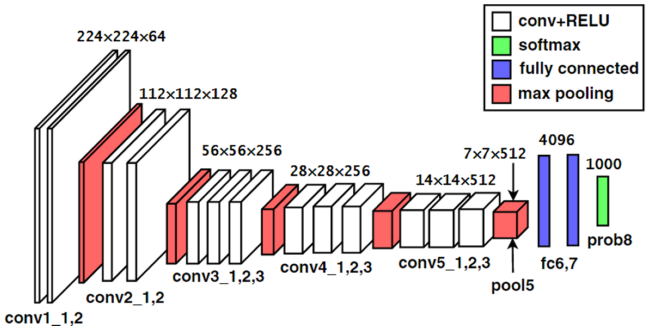


Fig. 2. VGG16 model architecture

In the proposed framework, the network is cut at a specific layer and the activations of the neurons in that layer are used as a representation of the input image. A simpler classifier is then applied over that feature space.

It has been proved that the outermost layers provide a more abstract and compact representation of the input images [12]. However, the final layers are more task-dependent and might not be useful if the target problem we want to address is very different from the problem the network was originally trained to solve. The layer that produces the most suitable representation for a given task must be determined empirically. To this end, the accuracy rates obtained with features from different layers and various classification methods are reported in section 3.

2.3 Final classifier

After the high level features have been extracted from the images using the deep neural network, a simpler classifier is in charge of emitting the final class prediction. Given that the features are sufficiently abstract, the use of a linear classifier is adequate. However, we also evaluated a simple nonparametric classifier that is not strictly a liner classifier, namely the k-Nearest Neighbor (kNN) algorithm.

In addition, the following linear classifiers were applied:

1. Support Vector Machine (C-SVM) with a linear kernel, as implemented in LIBSVM [1]. The decision function of this classifier is the following:

$$h(x) = sgn(w^T \varphi(x) + b) = sgn(w^T x + b)$$

Where w is calculated by solving the optimization problem presented in [2]. The multi-class support is obtained following a one-vs-one scheme.

2. Perceptron with linear activation function as implemented by *scikit-learn* [9]

$$h(x) = w^T x + b$$

Where w was optimized according to the mean squared error (mse) loss function. The stopping criteria was five iterations in all the experiments.

3. Logistic Regression (LR) as implemented in LIBLINEAR [3]. The decision function of this method is the following:

$$h(x) = \frac{1}{1 + e^{-\mu}} \; where \; \mu = w^T x + b$$

Here, the multi-class support was obtained using a one-vs-rest scheme. Note that despite the nonlinearity of the decision function, the decision boundary $\{x: h(x) = 0.5\}$ is a hyperplane and therefore this classifier is considered to be linear.

The majority of the web pages that we analyzed contained several images with relevant information. All the images must be taken into account to emit a prediction about the category of the web page. The simplest approach is taken where the final prediction is the most common label among the images of the web; when a tie occurs it is solved at random.

3 Experimental results

In this section, we evaluate the proposed method on a real word dataset for web page categorization. We provide insight into the suitability of transferred DCNN features by means of data visualization techniques and evaluate our system against individual image classification and web page categorization.

3.1 The dataset

To the best of our knowledge, there is no standard dataset for web page classification that focuses on visual content. For this reason we decided to collect a new database of web pages and the visual content present on them at the time. Our dataset consists of the images extracted from 75 web sites, uniformly distributed among five categories, namely “food and cooking”, “interior design”, “pets and wildlife”, “motor and races” and “fashion”. A total of 1,232 images were extracted from those web pages. The train/test split was arranged with 15 web pages for training and 60 web pages for testing purposes. As proved by our experimental results, the use of the transfer learning technique provides high accuracy rates in spite of the lack of a large training dataset.

Fig. 3. Class overlapping (interior design/pets and wildlife)

As expected, the artificial vision problem that emerged was of significant complexity. The dataset contains a high intra-class variability and some class overlapping (as shown in Figure 3).

3.2 DCNN feature visualization

In this section, we try to provide some insight into the suitability of the features at different depths of the network to solve the proposed problem. To do this, we use the t-distributed Stochastic Neighbor Embedding (t-SNE) algorithm [5]. This method is a powerful nonlinear dimensionality reduction algorithm and is generally used to visualize high-dimensional data. It works by iteratively minimizing a non-convex cost function so that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points. Due to the non-convexity of the cost function that is optimized with gradient descent, one may obtain slightly different results each time the algorithm is executed. If a given layer of the network produces a suitable representation, the activations of the neurons in that layer (obtained by feeding the network with the samples of our dataset) will form independent and easily separable clusters in the t-SNE 2D representation [5]. Figure 4 shows the 2D t-SNE embedding of the feature maps at different depths of the VGG16 model for all the images extracted from the collected web pages. Due to the similarity between the original problem VGG16 was trained to solve and the problem being addressed here, the features at deeper stages of the network produce t-SNE clusters with lower overlapping between classes. This suggests that features with a higher level of abstraction will propitiate better classification results.

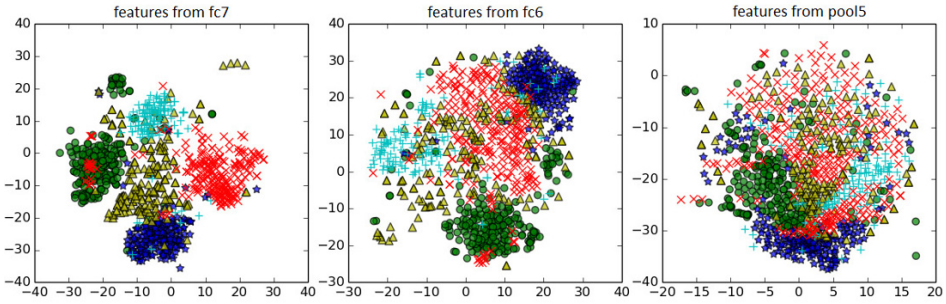


Fig. 4. t-SNE embedding of features at different depths. Food (●), motor (⋆), interior design (▲), animals (+) and fashion (×)

3.3 Classification accuracy results

In this section we report experimental results concerning the classification accuracy of both individual images and complete web pages. As described before, several classification methods were trained on the images extracted from 15 web pages, using the features at various depth levels of the network. The classification accuracy was then evaluated on the images extracted from the 60 remaining web sites. The accuracies for single image categorization are reported in Table 1. As explained before, the predictions for individual images are combined to categorize the complete web page. The accuracy rates for web page classification are shown in Table 2.

Table 1. Accuracy rates (%) on individual images

Features	SVM	LR	Perceptron	kNN
fc7	0.905	0.821	0.886	0.589
fc6	0.884	0.884	0.876	0.340
pool5	0.869	0.843	0.864	0.263
pool4	0.753	0.727	0.729	0.367
pool3	0.650	0.605	0.649	0.261
pool2	0.554	0.557	0.544	0.203

Table 2. Accuracy rates (%) on web categorization

Features	SVM	LR	Perceptron	kNN
fc7	0.966	0.9	0.93	0.56
fc6	0.916	0.95	0.933	0.31
pool5	0.9	0.85	0.916	0.25
pool4	0.816	0.816	0.766	0.4
pool3	0.766	0.583	0.766	0.25
pool2	0.733	0.55	0.7	0.2

4 Discussion and future work

In this paper, a novel framework for web page categorization was proposed. The system is able to classify web pages based on their visual content rather than their textual information. This makes the proposed technique very appropriate to modern web site analysis where visual elements have a dominant role. The major contribution of this paper is the application of transfer learning techniques to the problem of web page categorization. Experimental results show that this approach enables the construction of very accurate classifiers even when the artificial vision task to solve is significantly complex. In addition, our experiments show that competitive accuracy rates can be obtained with training phases of minutes, while training a complete deep neural network model to solve such a complex vision task typically requires hours or days, even if expensive specialized hardware is available. The second major advantage of the transfer learning approach is that it allows decent accuracy rates even if the training set is of

reduced length. The proposed approach could be further improved. First, a more sophisticated method to filter advertisements and other non-relevant content from web pages could be implemented. In addition, if more training data were available, it would be possible to adjust the weights of the deep neural network by executing various iterations of backpropagation. This would likely improve the accuracy rate of the proposed framework. Finally, a more advanced way of combining individual image predictions to categorize web pages could be developed, including techniques such as ensemble classification and mixture of experts [6].

Acknowledgements. The research of Daniel López-Sánchez has been financed by the University Faculty Training (FPU) program, reference number FPU15/02339.

5 References

[1] Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011). Software available at http://www.csie.ntu.edu.tw/~cjlin/

[2] Cortes, C., Vapnik, V.: Support-vector networks. Machine learning 20(3), 273–297 (1995)

[3] Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9, 1871–1874 (2008)

[4] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

[5] Maaten, L. V. D., & Hinton, G. (2008). Visualizing data using t-SNE. Journal of Machine Learning Research, 9 (Nov), 2579-2605.

[6] Masoudnia, S., & Ebrahimpour, R. (2014). Mixture of experts: a literature survey. Artificial Intelligence Review, 42(2), 275-293.

[7] Nair, V. G. (2014). Getting Started with Beautiful Soup. Packt Publishing Ltd.

[8] Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. IEEE Transactions on knowledge and data engineering, 22(10), 1345-1359.

[9] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825-2830.

[10] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115(3), 211-252.

[11] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[12] Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks?. In Advances in neural information processing systems (pp. 3320-3328).

Predicting the risk of suffering chronic social exclusion with machine learning

Emilio Serrano*, Pedro del Pozo-Jiménez , Mari Carmen Suárez-Figueroa, Jacinto González-Pachón, Javier Bajo, Asunción Gómez-Pérez
emilioserra@fi.upm.es

Ontology Engineering Group, Universidad Politécnica de Madrid, Spain

Abstract. The fight against social exclusion is at the heart of the Europe 2020 strategy: 120 million people are at risk of suffering this condition in the EU. Risk prediction models are widely used in insurance companies and health services. However, the use of these models to allow an early detection of social exclusion by social workers is not a common practice. This paper describes a data analysis of over 16K cases with over 60 predictors from the Spanish region of Castilla y León. The use of machine learning paradigms such as logistic regression and random forest makes possible a high precision in predicting chronic social exclusion. The paper is complemented with a responsive web available online that allows social workers to calculate the risk of a social exclusion case to become chronic through a smartphone.

Keywords: Social exclusion, Social services, Data analysis, Machine learning, Data mining.

1 Introduction

Social exclusion is a complex and multi-dimensional process involving the lack of resources, rights, goods and services, and the inability to participate in the normal relationships and activities, available to most people in a society, whether in economic, social, cultural or political scopes [8]. Social exclusion affects not only the quality of life of individuals, but also the equity and cohesion of society as a whole. The economic crisis is undermining the sustainability of social protection systems in the EU [1]: 24% of all the EU population (over 120 million people) are at risk of poverty or social exclusion [1]. The fight against poverty and social exclusion is at the heart of the Europe 2020 strategy for smart, sustainable and inclusive growth.

In chronic medical diseases, there is strong evidence supporting that early detection results in less severe outcomes. This paper intends to provide social workers with methods and tools to bring this early detection, which is so beneficial in the medical field, to the challenging problem of chronic social exclusion. To this purpose, the paper contributes with (1) an analysis of the social services

* ORCID ID: 0000-0001-7587-0703

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_16

data of Castilla y León (CyL), which is the largest region in Spain and counts with around two and a half million inhabitants. This analysis allows getting insights into why social exclusion can become chronic. Furthermore, a (2) machine learning model capable of quantifying the risk of chronic social exclusion is build. Finally, a (3) a responsive web is deployed to allow queries by social workers through a number of devices such as smartphones, tablets, or laptops. A RESTful web service is also provided to integrate the predictive capabilities into other software applications.

The paper outline is as follows. After revising some of the most relevant related works in section 2, the process used to analyze the data is explained in section 3. Section 4 reports the outcomes of the experiments conducted. Section 5 explains, analyzes, and compares the results. Section 6 introduces the web service implemented. Finally, section 7 concludes and offers future works.

2 Related works

Risk prediction models are widely used in insurance companies to allow customers to estimate their policies cost. Manulife Philippines [2] offers a number of online tools to calculate the likelihood of disability, critical illness, or death before the age of 65; based on age, gender, and smoking status. Health is another application field where risk estimations are undertaken for preventive purposes. More specifically, the risk of heart disease can be estimated at different websites such as at the Mayo clinic web [3].

There are a number of data analysis works in social exclusion that are detailed enough to extrapolate some of their methods to the research presented here. Ramos and Valera [12] use the *logistic regression* (LR) model to study social exclusion in 384 cases labeled by social workers through a heuristic procedure. Lafuente-Lechuga and Faura-Martínez [7] undertake an analysis of 31 predictors based on segmentation methods and LR. Haron [6] studies the social exclusion in Israel and proposes the *linear regression* as a better alternative to the LR. Suh et al. [13] analyze over 35K cases of 34 European countries using LR. Although these works are significant contributions to the social exclusion problem; they do not provide social workers with an online tool or an implemented machine learning model to cope with social exclusion as presented here. Moreover, the use of linear classifiers exclusively such as LR may hinder models from achieving a better predictive power.

3 Methodology

The methodology employed for this data analysis research is the widely used *Knowledge Discovery in Databases* (KDD) process described by Fayyad et al. [5]. This includes the following steps: Selection, Preprocessing, Transformation, Data Mining, and Evaluation and/or Interpretation. Although the KDD is an iterative and incremental process, some of the decision made in the different steps are presented unlooped here for the shake of clarity.

3.1 Selection

Eleven databases (DBs) with social services information were available to select relevant data. More specifically, the DBs were implemented with the Oracle object-relational database management system.

After several meetings with the social workers experts, 63 relevant variables from those DBs were selected to further study and preprocess. The predictors were identified by their use in different applications by the social workers. Nonetheless, locating these variables in the DBs to select them was specially challenging: a mapping from the variables to the DB (schema, table, and column) was not available. For example, the SAUSS application[1] has a schema with over 800 tables under the hood, plus a large number of tables shared among other applications.

Another important decision made in this step, after several iterations in the methodology, was the class definition to represent a chronic social exclusion sate. Several prediction services were outlined but the main class to study was defined as "having received social aid during 60 months or more", not necessarily continuously.

3.2 Preprocessing

The preprocess phase included among others: (1) the data integration where multiple data sources from the selection are combined; (2) the data cleaning removing noise and inconsistent data such as negative income or dates of birth in the year 1900; (3) managing missing values; and, (4) dealing with the temporal dimension.

Regarding the missing values, a number of variables whose values are missing in over 90% instances were not considered. Moreover, a clear positive correlation between missing data and non-chronic social exclusion was observed in the exploratory data analysis. This supports the idea that these missing values are not random but indicate that the social worker has decided not to log a particular measurement. Therefore, a special value of "NR" (not registered) has been included. As Witten et al. [14] explain, people analyzing medical databases have noticed that cases may be diagnosed simply from the missing values indicating tests that a doctor has decided not to make. Imputing values in these cases would result in an information loss.

3.3 Transformation

In this phase, data are transformed into forms appropriate for mining. This includes, among others: (1) standardization of numeric variables; (2) transforming internal numerical codes into interpretative nominal values; (3) aggregation for the multi-instance learning (where each labeled case in the data comprises several different instances); and, (5) dealing with the imbalanced classification problem.

[1] `https://sauss.jcyl.es/sauss-sso/`

The result of this process is a dataset with 63 predictors (some of the most important ones are described in section 5) and 16535 instances: 4205 of the positive class and 12330 of the negative class. This situation is known as imbalanced classification: a high accuracy is achieved by just predicting always the negative class. Some approaches to cope with this situation include: (1) penalized models; (2) undersampling the over represented class (negative);(3) oversampling the underrepresented class (positive); and, (4) generating synthetic samples. Section 4 shows several experiments in these lines.

3.4 Data mining

In this phase, machine learning paradigms are applied to create a hypothesis that explains the observations. The *logistic regression* (LR) is widely use to predict the risk of social exclusion as explained in related works, section 2. Furthermore, it is an intuitive solution when a class prediction is wanted with a degree of confidence. Experiments were also conducted, although not shown in this paper, using *decision trees* (which typically tolerate imbalanced data) and *rule-based classifiers* (whose hypotheses are highly interpretable for social workers). Meta-classifers such as *Boosting* and *Random Forests* (RF) were also considered given their higher predictive power. Besides, these are good of the box solutions that improve the maintenance when rebuilding new machine learning models in the light of new cases.

3.5 Evaluation

For the evaluation of the models, the cross validation is typically considered when the performance allows it. Using 10 folds involves rebuilding the machine learning model for the data 11 times. Nonetheless, when oversampling methods are applied, the cross validation method leads to overoptimistic results since the testing fold considers instances that are also present in the training folds. Thus, the classic partition between training and testing data was undertaken ensuring:

- the splitting (80/20% is considered) preserves the overall class distribution of the data (25% of positive cases);
- and, the oversampling is performed after this splitting both in the training and the testing data.

4 Results

Table 1 details the accuracy, precision, and recall for LR and RF. For the logistic regression, the experiments use a multinomial logistic regression model with a ridge estimator implemented in Weka [14]. This model requires adjusting the ridge parameter for underfitting or overfitting situations. For the random forest, the randomForest package of R [9] is employed with its defaults parameters, which include the use of 500 decision trees.

The table shows 9 different experiments including classification: (1) with the imbalanced data; oversampling the positive class with random sampling with replacement (2) before and (3) after splitting the testing data; oversampling the positive class with SMOTE [4] (4) before and (5) after separating the testing data; oversampling the positive class with ROSE [10] (6) before and (7) after splitting the testing data; and, undersampling the nagative class (8) by random sampling with replacement and by (9) the *K-medioids* [11] segmentation method.

	Logistic Regression			Random Forest		
Experiment	**Accuracy**	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**
Imbalanced	81.3%	69.3%	47.4%	80.6%	71.3%	40.3%
Oversampling 1	60.5%	78.5%	29%	91.5%	93.1%	89.7%
Oversampling 2	55.5%	78%	15.4%	67.8%	**88.6%**	40.9%
SMOTE 1	54.9%	82.1%	12.5%	89.2%	**96.1%**	81.8%
SMOTE 2	67.9%	72.6%	57.5%	82.4%	88.3%	74.7%
ROSE 1	58.8%	56.2%	73.4%	88%	90.7%	84.6%
ROSE 2	61.4%	59%	74%	73.4%	81.7%	57.4%
Undersampling 1	60.5%	62.7%	62.7%	75.2%	77.6%	70.9%
Undersampling 2	59.9%	59%	65.3%	74.6%	78.3%	68.1%

Table 1. Experiments results.

5 Discussion

The results show that for the imbalanced dataset the accuracy is reasonable but precision and recall are very low. This situation is known as *accuracy paradox*: the accuracy is only reflecting the underlying class distribution. Besides, the *precision*, considering the chronic social exclusion as positive class, is the most interesting metric because it allows social workers to find hazardous cases and to focus limited resources on them. Precision is a measure of quality as recall is a measure of quantity. High precision means: when the prediction says positive, the social worker can be very confident that it will be a chronic case.

Another clear result is that, as expected, the oversampling methods offer better results when the validation instances are extracted after oversampling the positive class. In "oversampling 1" this happens because several instances that are exactly the same are considered both for training and testing. When more advanced methods are employed such as SMOTE and ROSE, instead of duplicating cases, new instances are created by generalizating the points where the minority class is valid. Therefore, splitting a validation set after using SMOTE and ROSE is possible although it leads to optimistic results. Considering this, the two machine learning models selected are "oversampling 2" as a *conservative prediction*, and "SMOTE 1" as an *optimistic prediction*.

As explained above, the precision is the most interesting metric for the models purpose. In this vein, the selected conservative and optimistic models achieve

a high precision: 88.6% and 96.1%, respectively. These results are especially remarkable when compared to the imbalanced alternatives.

6 Implementation

A web service for predicting the risk of suffering chronic social exclusion based on the machine learning models explained has been deployed at: `http://webpact.oeg-upm.net/`. Due to usability reasons, instead of considering all the 63 predictors, only the ten most important variables are asked by the web service:

1. Age: calculated from day of birth to current date or date of death.
2. Level of studies: an ordinal indicator from illiterate to "higher education or vocational training".
3. Classification code: preliminary evaluation label given by the social workers whose values may be temporary, structural, undecided, or (as in most cases) unknown.
4. Annual income in euros: the training dataset contains a great deal of missing data for this variable, but it becomes highly relevant after imputing an average value.
5. Economic activity code: classified by the Spanish Ministry of Employment and Social Security.
6. Civil status.
7. Year of registration in local government.
8. Number of years as job seeker.
9. Professional qualification code: another indicator of education level ordering professional qualifications subject to recognition and accreditation. The code is given by the Spanish Ministry of Education, Culture and Sport.
10. National or foreigner: this binary variable (ternary with the "non registered" value) was obtained merging a number of nationality codes, most of them too unusual to offer a generic hypothesis about chronic social exclusion.

Limiting the attributes to 10, the accuracy, precision and recall of the model are 86.3%, 89.9% and 81.9%, respectively. More information about the model is provided online at `http://webpact.oeg-upm.net/info.html`.

Figure 1 shows, on the left, the presentation website and, on the right, the form to query the service about a case. The *Bootstrap* front-end web framework has been employed for designing the website and web application. This ensures the web responsiveness and allows social workers to access the service from a number of devices such as: computers, tablets, and smartphones.

The prediction returns a risk percentage for chronic social exclusion and it is considered a positive case when the risk is over 50%. The same queries may be conducted via RESTful service by introducing the query string parameters in the URL.

Fig. 1. A web service to detect risk of chronic social exclusion.

7 Conclusion and future works

This paper introduces a service to predict the risk of suffering chronic social exclusion with machine learning. With a precision around 90% in the most conservative predictions, it offers a quick rule of thumb that can detect citizens who are in danger of been excluded of the society beyond a temporary situation. The application is available online via web and RESTful web service and can be consulted by social workers from their smartphones. An early detection is possible thanks to this service and hence, as in medical diseases, the recovery process can be accelerated.

This service is based on an intelligent model that is fed with data from a whole Spanish region: eleven databases from the social services of Castilla y León. The classical Knowledge Discovery in Databases (KDD) process has been used and instantiated to the particularities of the data and application field. The results of the analysis reveal the age as the most relevant factor for chronic social exclusion. Besides, five of the top ten predictors are work or education related.

The future works in this research include but are not limited to: the use of deep learning techniques for feature extraction, the consideration of unlabeled cases to train supervised neural networks, the inclusion of more potentially relevant predictors in the studied observations, and the generation of new prediction models.

Acknowledgments

This publication would not have been possible without the inputs and collaboration of the Social Services of Castilla y León. This research work is supported by the "Junta de Castilla y León" under the public contract: "Servicios de elaboración de modelos matemáticos para realizar segmentación poblacional" (A2016/000271); by the EU Programme for Employment and Social Innovation (EaSI) under the project PACT ("people-oriented case management for social inclusion proactive model"); and, by the Spanish Ministry of Economy, Industry and Competitiveness under the R&D project Datos 4.0: Retos y soluciones (TIN2016-78011-C4-4-R, AEI/FEDER, UE).

References

1. European Commission's DG for Employment, Social Affairs & Inclusion. `http://ec.europa.eu/social/main.jsp?catId=751`. Accessed: February of 2017.
2. Manulife Philippines. Calculate your risk, your partner's risk or both. `ttp://www.insureright.ca/what-is-your-risk`. Accessed: February of 2017.
3. Mayo Clinic. Heart Disease Risk Calculator. `http://www.mayoclinic.org/diseases-conditions/heart-disease/in-depth/heart-disease-risk/itt-20084942`. Accessed: February of 2017.
4. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1):321–357, June 2002.
5. U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. Advances in knowledge discovery and data mining. chapter From Data Mining to Knowledge Discovery: An Overview, pages 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996.
6. N. Haron. *On Social Exclusion and Income Poverty in Israel: Findings from the European Social Survey*, pages 247–269. Springer US, Boston, MA, 2013.
7. M. Lafuente-Lechuga and U. Faura-Martínez. An álisis de los individuos vulnerables a la exclusión social en españa en 2009. *Anales de ASEPUMA*, (21), 2013.
8. R. Levitas, C. Pantazis, E. Fahmy, D. Gordon, E. Lloyd, and D. Patsios. *The multidimensional analysis of social exclusion.* London: Social Exclusion Task Force, Cabinet Office, 2007.
9. A. Liaw and M. Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.
10. N. Lunardon, G. Menardi, and N. Torelli. Rose: A package for binary imbalanced learning. *R Journal, The*, 6(1):79–89, 2014.
11. H.-S. Park and C.-H. Jun. A simple and fast algorithm for k-medoids clustering. *Expert Syst. Appl.*, 36(2):3336–3341, Mar. 2009.
12. J. Ramos and A. Varela. Beyond the margins: Analyzing social exclusion with a homeless client dataset. *Social Work & Society*, 14(2), 2016.
13. E. Suh, P. TiffanyVizard, and T. AsgharBurchardt. Quality of life in europe: Social inequalities. *3rd European Quality of Life Survey*, 2013.
14. I. H. Witten, E. Frank, and M. A. Hall. *Data mining: practical machine learning tools and techniques.* Morgan Kaufmann, Burlington, MA, 2011.

Semantic Profiling and Destination Recommendation based on Crowd-sourced Tourist Reviews

Fátima Leal[1,4], Horacio González–Vélez[2], Benedita Malheiro[3,4], and Juan Carlos Burguillo[1]

[1] ETSET/UVigo – School of Telecommunications Engineering, University of Vigo, Vigo, Spain
[2] CCC/NCI – Cloud Competency Centre, National College of Ireland, Dublin, Ireland
[3] ISEP/IPP – School of Engineering, Polytechnic Institute of Porto, Porto, Portugal
[4] INESC TEC, Porto, Portugal
fatimaleal2@gmail.com, horacio@ncirl.ie, mbm@isep.ipp.pt, J.C.Burguillo@uvigo.es

Abstract Nowadays tourists rely on technology for inspiration, research, booking, experiencing and sharing. Not only it provides access to endless sources of information, but has become an unbounded source of tourist-related data. In such crowd-sourced data-intensive scenario, we argue that new approaches are required to enrich current and new travelling experiences. This work, which supports the "dreaming stage", proposes the automatic recommendation of personalised destinations based on textual reviews, *i.e.*, a semantic content-based filter of crowd-sourced information. Our approach relies on Topic Modelling – to extract meaningful information from textual reviews – and Semantic Similarity – to identify relevant recommendations. Our main contribution is the processing of crowd-sourced tourism information employing data mining techniques in order to automatically discover untapped destinations on behalf of tourists.

Keywords: Tourism; Crowdsourcing; Topic Modelling; Profiling; Recommendation

1 Introduction

Over the last two decades, the Internet has increased the accessibility of travelling. Tourists not only plan trips based on websites, but, in fact, employ technologies throughout the travel cycle. According to the World Tourism Organization [16], the travel cycle encompasses the dreaming, researching, booking, experiencing and sharing stages. It all starts with the dreaming stage, when the tourist starts to consider travelling, and proceeds with the research stage, when the tourist invests time searching for options. Once the tourist makes his/her mind, the booking stage begins. Finally, in the experiencing stage, the tourist embarks on the trip and relies on context-aware mobile applications to get personalised recommendations. Last, but not least, in the sharing stage, the tourist shares feedback data both in real and deferred time.

This pervasive interaction between tourists and technology consistently generates large volumes of collaborative information on dedicated platforms. Arguably, tourists build their own "crowd-sourced" profile, as their personal information is directly

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_17

entered or harvested from tourism websites. Overall, this process comprises experience sharing in the form of likes, posts, images and/or videos (social-network-based); ratings and reviews (evaluation-based); and pages (wiki-based). This valuable feedback information ultimately influences the decisions of both tourists and businesses.

Although tourism crowd-sourced information influences decision making, typically, a tourist cannot monitor or control his/her own crowd-sourced footprint to enhance his/her options, due to the complexity of the diverse platforms and resources. As machine learning and data mining methodologies provide dedicated algorithms for knowledge discovery, we argue that the combination of tourism crowd-sourced information and machine learning methodologies should further enable the personalisation of the tourist travel cycle, eventually proposing *ad hoc* travel stages based on the tourist crowd-sourced footprint.

This paper explores the use of tourism crowd-sourced information to enhance the travel cycle stages. Specifically, we have designed an algorithm to recommend personalised destinations based on Expedia crowd-sourced hotel textual reviews, *i.e.*, for the dreaming stage. Our contribution automatically combines tourism crowd-sourced information and recommender systems to discover untapped destinations for tourists, employing both Topic Modelling (TM) to extract meaningful information from textual reviews; and Semantic Similarity (SS) to recognise relevant recommendations.

Our technique applies content-based filtering to topic-modelled tourists and locations. Topics are clusters of words aggregated according to their meaning as well as frequency in the textual reviews. The content-based filtering provides recommendations according to the semantic similarity among topics. The recommendation engine is assessed through evaluation metrics, *i.e.*, Precision, Recall, and F-Measure.

This paper is organised as follows. Section 2 reviews related work on recommendation supported by crowdsourcing. Section 3 describes the proposed method, including the description of the algorithms used. Section 4 presents the implementation details. Section 5 reports the experiments performed and the results obtained. Finally, Section 6 provides the conclusions and discusses the outcomes of this work.

2 Related Work

There is a significant number of well-known Web-based tourism portals (*e.g.*, TripAdvisor, Expedia, airbnb, Wikivoyage, *etc*.) where a tourist can search, comment, share, and evaluate resources. These collaborative platforms can be envisaged as reputation-based crowdsourcing platforms, as users can increase their reputation by evaluating and making recommendations. Overall, they enable tourists to build their own digital footprint and implement profiling and recommendation mechanisms, i.e. tourists themselves contribute their digital foorprints to form intelligent "crowd-sourced" recommendation systems [4].

On the one hand, recommendation systems for tourism have been extensively studied in the literature. Borras et al. [3], Gavalas et al. [6], and Felfernig et al. [5] provide comprehensive surveys on the recommendation of tourism resources. Some systems just harvest public portal information to suggest destinations or to plan trips [13], while others propose the aggregation of tourism-related information in the context of user

models for personalisation [7]. Of particular relevance to our work is Patil et al. [12] which have surveyed frequent data mining techniques used by tourism recommendation systems. Although some of the tourism recommendation systems documented in the literature use machine learning algorithms to detect tourist preferences, frequent behaviours, new trends, or contexts, most systems do not extensively employ crowd-sourced information.

On the other hand, Tiwari and Kaushik [15], Bachrach et al. [1], and Zhuang et al. [18] use a questionnaire/form-based approach to collect crowd-sourced information. Tiwari and Kaushik specifically rely on the crowd available *in situ* to get updated information and, thus, enrich the list of recommendations. Bachrach et al. ask tourists to rate a set of 20 attractions in order to predict the overall crowd-sourced rating for each attraction. Zhuang et al. provide a form for tourists to suggest each other travel plans. Nonetheless, such questionnaire/form-based crowdsourcing approach requires additional interaction from the tourists, disregarding existing tourism crowdsourcing platforms. Yu et al. [17] collect data from Location-Based Social Networks (LBSN) to model users and, thus, recommend destinations. Finally, Guo et al. [9] combine data from different crowdsourcing sources, but do not provide recommendations. In particular, they get popular routes from travelogues and build scenic spots by matching photo descriptions with attraction reviews.

2.1 Contribution

Scant research has been devoted to the automatic use of heterogeneous crowd-sourced information in tourism recommendation systems fully employing data mining techniques. Table 1 depicts a comparison of the above mentioned related work. While earlier work requires additional interaction by the tourists (questionnaire/forms) or does not provide recommendations (*e.g.*, heterogeneous sources), we are building upon our previous work [10], which relies solely on the available crowd-sourced data. We have explored both tourism crowd-sourced data and recommendation techniques, employing data mining methods, in order to discover untapped destinations for tourists. This research extends this approach by processing a significant amount of heterogeneous crowd-sourced information via tourist reviews to improve tourist recall and destination/location features.

Table 1. Comparison of tourism crowdsourcing and recommendation systems

Systems	Reviews-based Modelling	Crowdsourcing Modality	Data Mining	Recommendation
Tiwari and Kaushik (2014)	No	Questionnaire	No	Context-aware
Bachrach et al. (2014)	No	Questionnaire	No	Collaborative
Zhuang et al. (2014)	No	Form	No	Collaborative
Yu et al. (2016)	No	LBSN	No	Collaborative
Guo et al. (2016)	No	Heterogeneous	Yes	–
Leal et al. (2017)	**Yes**	**Form (Expedia)**	**Yes**	**Content-based**

3 Proposed Method

Our approach automates the dreaming stage by recommending new locations to users based on crowd-sourced information gathered from the Expedia platform. Algorithm 1 summarises the recommendation engine. The algorithm accepts as inputs the required LDA parameters as well as hotel and tourist data, including the crowd-sourced textual reviews, and outputs a list of recommended locations for each tourist.

Algorithm 1 Recommendation algorithm

Inputs	Hotel Data: $H = (<\text{Hotel } h, \text{Location } l>, \ldots)$ User Data: $U = (<\text{User } u, \text{Hotel } h, \text{Review } r_{u,h}>, \ldots)$ LDA Parameters: θ; β; n_{iter} and n_{topics}
Outputs	Ordered List of Recommended Locations per User: $L_u = (l_a, \ldots, l_n)$
Step 1	**Review Data Preprocessing**: Tokenising, Stopping & Aggregation
Step 2	**Parallel Topic Modelling** via LDA: **for** $(u = 0; u < users; u{+}{+})$ **do** $topics_u \leftarrow GetUserTopics()$ **for** $(l = 0; l < locations; l{+}{+})$ **do** $topics_l \leftarrow GetLocationTopics()$

Step 3	**Recommendations**:
	for $(u = 0, u < users; u{+}{+})$ **do**
	  for $(l = 0; l < locations; l{+}{+})$ **do**
	  $SS_{u,l} \leftarrow GetTopicSimilarity(topics_u; topics_l)$
	for $(u = 0, u < users; u{+}{+})$ **do** return locations sorted by $SS_{u,l}$

Step 4	**Evaluation Procedure**: $Precision$, $Recall$ and $F\text{-}measure$

Data Preprocessing tokenises textual reviews, including individual user as well as the aggregated location reviews, and removes meaningless words (special characters, stop words, and numerical parameters).

Topic Modelling aims to find patterns in unstructured texts, attempting to inject semantic meaning into the vocabulary. Topic modelling algorithms represent a set of computer programs which extract topics from texts. A topic is a list of words which occurs in statistically meaningful ways [8]. We employed Latent Dirichlet Allocation (LDA) topic modelling to extract keywords and use them as implicit features of the respective users and locations. LDA, while a Bayesian generative model for text structures, sees a text corpus (d) as a collection of t topics, where a topic has a probability distribution (θ_d, $d = 1$ to k) over a word dictionary (Equation 1).

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\Pi_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \ldots \theta_k^{\alpha_k - 1} \tag{1}$$

The α is k-vector with elements $\alpha_i > 0$ and $\Gamma(x)$ is the gamma function. Now, let β_t be the multinomial distribution over words for topic t. Thus, each word is assigned to a topic via β_t distribution. Blei *et al.* (2003) contains a complete mathematical description regarding LDA algorithm [2]. In our problem, we consider each user review as a document. The underlying topics are the set of possible implicit features.

Semantic Similarity measures the distance between concept meanings. The semantic relatedness is computed using ontologies to measure the distance between terms or concepts. One of the most well-known resources which have been extensively used to compute semantic relatedness is WordNet. This paper uses semantic similarity of Princeton's WordNet to calculate the distance between location and tourist topics identified via topic modelling. WordNet (`http://wordnet.princeton.edu`) is a lexical ontology of English words which organises names, adjectives, verbs, and adverbs according to semantic relations (synonymy, antonymy, hyperonymy and meronymy) [11].

Evaluation Metrics The results were assessed using the Precision Recall and F-Measure classification metrics. On the one hand, the Precision measures the proportion of good recommendations (quality). On the other hand, the Recall measures the proportion of good recommendations which appear among the most important recommendations (quantity). Finally, F-Measure combines both metrics [14].

4 Implementation

Our recommendation system, which is implemented in Java, runs on an Openstack cloud instance with 16 GB RAM, 8 CPU and 160 GB hard-disk. The Java-based application relies on the MAchine Learning for LanguagE Toolkit (`http://mallet.cs.umass.edu/`) (MALLET) for parallel topic modelling and on the WordNet Similarity for Java (WS4J) library (`http://ws4jdemo.appspot.com`) to compute semantic relatedness. In terms of architecture, our content-based filter, which is depicted in Figure 1, includes four modules: (*i*) Data Preprocessing; (*ii*) Topic Modelling; (*iii*) Semantic Similarity; and (*iv*) Recommendation. First, Data Preprocessing removes irrelevant words and aggregates user and location reviews. Then, Topic Modelling applies LDA to identify the most representative topics related with tourists and locations (MALLET). Next, the Semantic Similarity relies on WordNet to compare semantically tourists and location topics. Finally, it orders and recommends, for each user, the locations by descending relatedness.

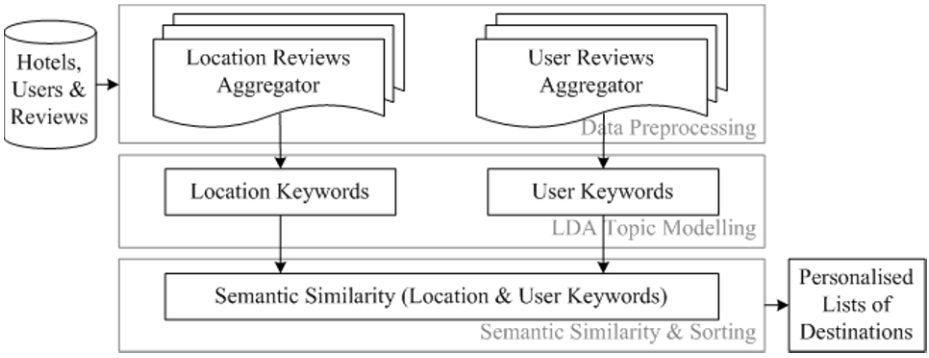


Figure 1. Recommendation engine

5 Experiments and Results

We conducted off-line experiments to evaluate the effectiveness and usefulness of the proposed method with the Expedia crowd-sourced data.

The HotelExpedia data set (`http://ave.dee.isep.ipp.pt/~1080560/ExpediaDataSet.7z`) contains 6030 hotels, 3098 reviewers, including anonymous reviewers, and 381 941 reviews from 11 different hotel locations. Although the data set includes anonymous reviewers and their reviews, we only used in our experiments crowd-sourced data from the 1089 identified reviewers, *i.e.*, we discarded the anonymous users and their inputs. Each user introduced at least 20 reviews. The data set was randomly partitioned into training (75 %) and test (25 %). This data set was built and used as a case study of crowdsourcing in the tourism domain.

5.1 Recommendation Engine

The experiments were focussed on the Parallel Topic Modelling, Recommendation and Evaluation Procedure.

Parallel Topic Modelling is the core of our recommendation engine. It analyses large volumes of textual reviews in order to find topics for describing locations and users. A topic is a cluster of words which, frequently, occur together. Moreover, our topic modelling implementation connects words with similar meanings and distinguishes words with multiple meanings. For each location and user, we select 10 topics using 1000 iterations. The number of topics and the number of iterations were selected according to the F-Measure of the recommendations. The algorithm attributes to each topic a weight based on its relevance in the document. Table 2 and Table 3 present the top five topics together with the corresponding weights (W) for Barcelona and User 15.

Table 2. Barcelona topics

Topic	W	Words
1	0.19	noise night air noisy street
2	0.13	shower bathroom water small door
3	0.34	great close walking clean nice
4	0.17	desk front day time check
5	0.33	stay great friendly clean excellent

Table 3. User 15 topics

Topic	W	Words
1	0.28	check time desk day told
2	0.60	stay good great friendly comfort
3	0.15	airport bus good terminal flight
4	0.46	tube station walk great close
5	0.31	small bathroom shower nice air

Recommendations are based on the WordNet semantic relatedness between the topics provided by the parallel topic modelling module. Considering the example of Table 2 and Table 3, there are similarities between: (*i*) Topic 5 from Barcelona and Topic 2 from User 15; and (*ii*) Topic 2 from Barcelona and Topic 5 from User 15. The algorithm then computes the weighted average between the related topics, using the corresponding topic weight. In the end, the system suggests future travelling destinations to tourists by choosing, for each tourist, the locations with the higher semantic similarity.

Evaluation Procedure adopts Precision, Recall and F-Measure metrics to assess the quality of recommendation. Figure 2 plots, for different number of topics per entity, the F-Measure results. The values grow from 1 to 10 topics and, then, decrease from

10 till 20 topics. The best F-Measure value was achieved with 10 topics per entity, *i.e.*, when the system uses 10 topics to represent tourists and locations. In this case, the recommendation engine presents an F-Measure of 78 %. Based on these results, Figure 3 plots the F-Measure and runtime versus the number of Topic Modelling iterations when using 10 topics per entity. The best F-Measure value was achieved with 1000 iterations and a total runtime of 90 ks (25 h).

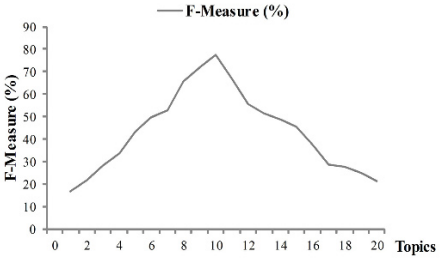


Figure 2. F-Measure *vs.* topics

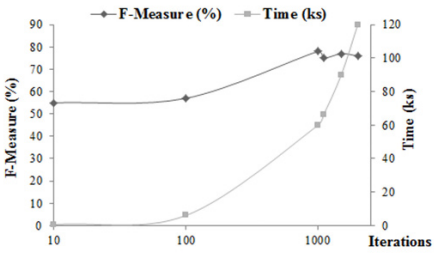


Figure 3. F-Measure and runtime *vs.* iterations

6 Conclusions

Technology has revolutionised both travelling and the tourism industry. In particular, not only it assists tourists in all stages of their travelling, including researching, booking, experiencing as well as sharing, but has transformed tourism business oriented platforms into crowdsourcing platforms, *e.g.*, Expedia, where tourism related knowledge accumulates as tourists leave their digital footprints. These digital footprints are highly influential for businesses and tourists alike.

This paper explores tourism crowd-sourced information to enrich the dreaming stage of the travel cycle. In terms of contributions, we use tourism crowd-sourced textual information and data mining methods to discover and recommend untapped destinations for tourists. Our approach, which uses textual reviews from Expedia, profiles users and locations based on LDA topic modelling and produces personalised recommendations regarding future user destinations based on semantic topic similarity. In order to achieve better recommendations, we tested the topic modelling module with different number of topics and, then, with different number of iterations. The resulting content-based filter was able to recommended future destinations to users solely based on textual reviews with an F-Measure of 78 %.

As future work, we intend to: (*i*) increase the data set dimension with more locations/reviews to explore the tourism Big Data concept in crowdsourcing platforms; (*ii*) introduce hotel recommendations; and (*iii*) design a trust and reputation model for assessing the reliability of the review publishers.

7 Acknowledgements

This work was partially financed by: (*i*) the European Regional Development Fund (ERDF) through the Operational Programme for Competitiveness and Internationalisation - COMPETE Programme - within project ≪FCOMP-01-0202-FEDER-023151≫ and

project ≪POCI-01-0145-FEDER-006961≫, and by National Funds through Fundação para a Ciência e a Tecnologia (FCT) - Portuguese Foundation for Science and Technology - as part of project UID/EEA/50014/2013; and (*ii*) ICT COST Action IC1406 High-Performance Modelling and Simulation for Big Data Applications (cHiPSet).

References

1. Bachrach, Y., Ceppi, S., Kash, I.A., Key, P., Radlinski, F., Porat, E., Armstrong, M., Sharma, V.: Building a personalized tourist attraction recommender system using crowdsourcing. In: AAMAS '14. pp. 1631–1632. International Foundation for Autonomous Agents and Multiagent Systems, Paris (2014)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of machine Learning research 3, 993–1022 (2003)
3. Borras, J., Moreno, A., Valls, A.: Intelligent tourism recommender systems: A survey. Expert Systems with Applications 41(16), 7370–7389 (2014)
4. Chittilappilly, A.I., Chen, L., Amer-Yahia, S.: A survey of general-purpose crowdsourcing techniques. IEEE Transactions on Knowledge and Data Engineering 28(9), 2246–2266 (2016)
5. Felfernig, A., Gordea, S., Jannach, D., Teppan, E., Zanker, M.: A short survey of recommendation technologies in travel and tourism. OEGAI Journal 25(7), 17–22 (2007)
6. Gavalas, D., Kasapakis, V., Konstantopoulos, C., Mastakas, K., Pantziou, G.: A survey on mobile tourism recommender systems. In: ICCIT 2013. pp. 131–135. IEEE, Beirut (2013)
7. Gonzalez, G., Lopez, B., De la Rosa, J.: Smart user models for tourism: A holistic approach for personaliaed tourism services. Information Technology & Tourism 6(4), 273–286 (2003)
8. Graham, S., Weingart, S., Milligan, I.: Getting started with topic modeling and mallet. The Programming Historian 2, 12 (2012)
9. Guo, T., Guo, B., Zhang, J., Yu, Z., Zhou, X.: Crowdtravel: Leveraging heterogeneous crowdsourced data for scenic spot profiling and recommendation. In: PCM 2016. LNCS, vol. 9917, pp. 617–628. Springer, Xian (2016)
10. Leal, F., Dias, J.M., Malheiro, B., Burguillo, J.C.: Analysis and visualisation of crowdsourced tourism data. In: C3S2E '16. pp. 98–101. ACM, Porto (2016)
11. Miller, G.A.: WordNet: A lexical database for English. Commun. ACM 38(11), 39–41 (Nov 1995)
12. Patil, P., Kolhe, V.: Survey of travel package recommendation system. International Journal of Science and Research 3(12), 1557–1561 (2014)
13. Ricci, F., Werthner, H.: Case base querying for travel planning recommendation. Information Technology & Tourism 4(3-1), 215–226 (2001)
14. Shani, G., Gunawardana, A.: Evaluating recommendation systems. In: Recommender systems handbook, pp. 257–297. Springer (2011)
15. Tiwari, S., Kaushik, S.: Crowdsourcing based fuzzy information enrichment of tourist spot recommender systems. In: ICCSA 2015. LNCS, vol. 9158, pp. 559–574. Springer, Banff (2015)
16. World Tourism Organization (UNWTO) Affiliate Members: Technology in tourism. AM-reports 1, UNWTO (2011)
17. Yu, Z., Xu, H., Yang, Z., Guo, B.: Personalized travel package with multi-point-of-interest recommendation based on crowdsourced user footprints. IEEE Transactions on Human-Machine Systems 46(1), 151–158 (2016)
18. Zhuang, Y., Zhuge, F., Chiu, D., Ju, C., Jiang, B.: A personalized travel system based on crowdsourcing model. In: ADMA 2014. LNCS, vol. 8933, pp. 163–174. Springer, Guilin (2014)

Robustness of Coordination Mechanisms in Distributed Problem Solving against Erroneous Communication

Friederike Wall

Alpen-Adria-Universitaet Klagenfurt, 9020 Klagenfurt, Austria
friederike.wall@aau.at,
WWW home page: http://www.aau.at/csu

Abstract. This paper investigates how robust coordination mechanisms employed by distributed problem solving systems are against communication errors. For this, an agent-based simulation based on the framework of NK fitness landscapes is applied. The study controls for different levels of erroneous communication and of interdependencies among the distributed problems to be solved. The results provide broad, though not universal support for detrimental effects of erroneous communication. Moreover, the results indicate that the complexity of the overall problem to be solved considerably shapes the sensitivity of coordination mechanisms to erroneous communication.

Keywords: agent-based simulation, communication errors, complexity, coordination, NK fitness landscapes

1 Introduction

The coordination of autonomous agents' actions with respect to the overall systems' objective in Distributed Problem Solving (DPS) is an interdisciplinary field of research employing ideas from various disciplines, like computer science, organization theory or economics to name but a few [7]. Coordination is of particular relevance when interrelations among the agents' tasks occur (e.g., [2], [8], [14], [15]). Whether a system of distributed problem solvers works as a unit, i.e. in a coordinated manner, is captured by metrics of coherence in terms of certain systems properties (e.g., solution quality) [2].

Many mechanisms of coordination employ direct communication among the agents, especially, for agents informing each other about the intended actions and, eventually, negotiating and mutually adjusting their plans (plan revision) [8], [11]. However, communication among agents does not necessarily need to be perfect [3], [9]. Hence, an interesting research question is the following: *How robust are mechanisms for coordinating plans of autonomous agents against erroneous communication of intended plans and, especially, which solution quality is achieved in the course of plan coordination that is affected by erroneous communication?*

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_18

This paper intends to contribute to that research question and, for this, it employs an agent-based simulation based on the framework of NK fitness landscapes [4], [5]. In particular, the distributed systems studied consist of autonomous agents where each is responsible for searching for solutions of a certain sub-task; there is no central planning authority; though agents are not hostile against each other, each agent pursues its own objective; the agents communicate and, eventually, modify their plans (plan revision); the communication errors investigated are not intentionally (in order to achieve a certain private benefit) built in by the sending agents, but do occur accidentally. The simulation experiments control for different levels of communication errors and of task complexity which affects the coordination needs.

2 Outline of the Simulation Model

2.1 Distributed Problem and Task Assignment

The distributed systems simulated consist of M agents indexed by $r = 1, ..., M$. The agents collaboratively search for superior solutions of an N-dimensional binary search problem which is partitioned into M disjoint N^r-dimensional sub-problems. Each sub-problem is assigned to one agent r. In each time step t of the search, each agent makes a plan regarding its partition of the overall problem. The coordination mode employed determines to which other agents $q \neq r$ agent r communicates its plan and how the plans may be adjusted.

The N-dimensional binary search problem follows the framework of NK-fitness landscapes as originally introduced in the domain of evolutionary biology [4] and, since then, broadly employed (for an overview see [13]). From a more "technical" point of view NK landscapes are stochastically generated pseudo-boolean functions with N bits, i.e., $F : \{0,1\}^N \rightarrow R^+$. A major feature of NK fitness landscapes is that they allow to easily control for the complexity of the N-dimensional search problem as captured by parameter K [1], [6]. In particular, in each time step a system of distributed search agents seeks for a superior configuration $\boldsymbol{d_t}$ of an N-dimensional binary problem, i.e. $\boldsymbol{d_t} = (d_{1t}, ..., d_{Nt})$ with $d_{it} \in \{0,1\}$, $i = 1, ...N$, out of 2^N different binary vectors possible. Each of the two states $d_{it} \in \{0,1\}$ contributes with C_{it} to overall system's performance $V(\boldsymbol{d_t})$. In line with the NK framework, C_{it} is randomly drawn from a uniform distribution with $0 \leq C_{it} \leq 1$. The parameter K (with $0 \leq K \leq N-1$) reflects the number of those choices d_{jt}, $j \neq i$ which also affect the performance contribution C_{it} of choice d_{it}. Hence, contribution C_{it} may not only depend on the single choice d_{it} but also on K other choices:

$$C_{it} = f_i(d_{it}; d_{i_1 t}, ...d_{i_K t}), \tag{1}$$

with $\{i_1, ..., i_K\} \subset \{1, ..., i-1, i+1, ..., N\}$.

In case of no interactions among choices, K equals 0, and K is $N-1$ for maximum interactions where each single choice i affects the performance contribution of each other binary choice $j \neq i$. The overall performance V_t achieved in

period t results as normalized sum of contributions C_{it} from

$$V_t = V(\boldsymbol{d_t}) = \frac{1}{N}\sum_{i=1}^{N} C_{it} \tag{2}$$

In each time step t, an agent seeks to identify the best configuration for the "own" choices $\boldsymbol{d_t^r}$ assuming that the other agents do not alter their prior choices. Agent r randomly discovers two alternatives to status quo $\boldsymbol{d_{t-1}^{r*}}$ - one that differs in one choice ($a1$) and another ($a2$) where two choices are modified compared to the status quo. Hence, in time step t, agent r has three options to choose from: to keep the status quo or to switch to $\boldsymbol{d_t^{r,a1}}$ or $\boldsymbol{d_t^{r,a2}}$. For the formation of preferences at time step t, an agent r assesses these options according to their effects on the agent's objective which is given by:

$$P_t^r(\boldsymbol{d_t^r}) = \frac{1}{N}\sum_{i=1+w}^{N^r} C_{it} \tag{3}$$

where $w = \sum_{s=1}^{r-1} N^s$ for $r > 1$ and $w = 0$ for $r = 1$. Next, agent r compiles a list of preferences $L_t^r = \left\{\boldsymbol{d_t^{r,p1}}, \boldsymbol{d_t^{r,p2}}, \boldsymbol{d_t^{r,p3}}\right\}$ where $\boldsymbol{d_t^{r,p1}}$ indicates the most preferred option according to the agent's objective (Eq. 3), and so on. Hence, as a result of the agents' search and assessment of options, each agent r has an ordered list L_t^r of preferences which we call agent r's plan.

2.2 Erroneous Communication

In very center of this study is erroneous communication occurring in the course of coordination - may it be due to errors on the senders' site, imperfect transmission of plans or erroneous perception on the receivers' sites. However, for the sake of simplicity, in this study it is assumed that the erroneously communicated options of a plan are the same for all the recipients of that plan (which is particular reasonable if communicative faults occur on the sender's site). The level of erroneous communication is captured by the number E^C of single choices i out of the N-dimensional search problem which are not communicated correctly in the course of coordination.

In particular, after the agents have made their plans for period t as captured in $L_t^r \forall r \in \{1, \cdots M\}$ and before they inform each other about their plans, a list Q_t with $|(Q_t)| = E^C$ entries is generated randomly. The list Q_t contains those single choices (bits) i out of the N dimensions of the binary search problem which will be flipped in the communication among agents in time step t. For example, if the error level is $E^C = 3$ and $Q_t = \{2, 6, 12\}$ the second, the sixth and the twelfth bit will be flipped when it comes to communication of the agents' plans. (Recall that the choice of each i is unambiguously assigned to one of the M agents.) Each of the N single choices has the same probability to be chosen for list Q_t.

Hence, before it comes, eventually, to the revision of plans the "true" plans L_t^r are distorted according to Q_t. The resulting distorted plan $\widetilde{L}_t^r$ is a function of the "true" plans and the list of distortions Q_t:

$$\widetilde{L}_t^r = \widetilde{L}_t^r(L_t^r, Q_t) \tag{4}$$

with $\widetilde{L}_t^r$ capturing "distorted" options $\widetilde{L}_t^r = \left\{ \widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{r,p1}}, \widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{r,p2}}, \widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{r,p3}} \right\}$.

2.3 Coordination Mechanisms

The simulation model includes two modes of coordination: (1) The *sequential mode* captures what is also named "sequential planning". The agents make their choices sequentially where, for the sake of simplicity, the sequence is given by the index r of the agents. In particular, in time step t agent r with $1 < r \leq M$ is, eventually imperfectly, informed by agent $r-1$ about the choices made so far, i.e., made by the "preceding" agents $< r$. Agent r takes these choices into account and re-evaluates its "true" own options $\boldsymbol{d}_{\boldsymbol{t-1}}^{\boldsymbol{r*}}, \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,a1}}$ and $\boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,a2}}$ according to Eq. 3 – potentially resulting in an adjusted list of preferences $L_t^r = \left\{ \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p1}}, \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p2}}, \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p3}} \right\}$ and chooses $\boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p1}}$. $\boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p1}}$ here depends on $\widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{r-1,p1}}$, i.e., is a function of the choices of the preceding agents and distorted according to Q_t. Obviously, only agent 1 does not have to take a previous choice into account. Hence the choice of agent r is made according to

$$\boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r*}} = \begin{cases} \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p1}} \left(\widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{r-1,p1}} \right) & \text{for } 2 \leq r \leq M \\ \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p1}} & \text{for } r = 1 \end{cases} \tag{5}$$

and with Eq. (5) the overall configuration results from $\boldsymbol{d_t} = \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{M*}}$.

(2) In the *horizontal veto mode*, first, the agents mutually – and potentially erroneously – inform each other about their most preferred options $\widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{r,p1}}$. Next, each agent r evaluates the composite vector $\boldsymbol{d^{rC}}$ according to Eq. 3. $\boldsymbol{d^{rC}}$ consists of the – potentially distorted – first preferences of the other agents $q \neq r$ and the "true" own preference, i.e. $\boldsymbol{d^{rC}} = (\widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{1,p1}}, ... \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p1}}, ... \widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{M,p1}})$. However, the agents are endowed with mutual veto power: since no agent accepts a worse result than achieved in the previous period, the composite vector $\boldsymbol{d^{rC}}$ requires to, at least, promise the partial performance $P_t^r(\boldsymbol{d_{t-1}})$ of the status quo to each single agent r. Hence, to be accepted the following condition has to be fulfilled:

$$P_t^r(\boldsymbol{d^{rC}}) \geq P_t^r(\boldsymbol{d_{t-1}}) \;\; \forall r \in \{1, ... M\} \tag{6}$$

If this condition is met for each agent r, the first "true" preferences $\boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p1}}$ of each agent is implemented. In contrast, if, at least, one agent vetoes then the agents' second preferences – eventually distorted according to Q_t – by each agent are composed to $\boldsymbol{d^{rC}} = (\widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{1,p2}}, ... \boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p2}}, ... \widetilde{\boldsymbol{d}}_{\boldsymbol{t}}^{\boldsymbol{M,p2}})$, evaluated according to Eq. 3 and assessed in regard of Eq. 6. If at least one agent vetoes (again, since the condition in Eq. 6 is not met), then the status quo is kept, i.e., $\boldsymbol{d_t} = \boldsymbol{d_{t-1}}$, otherwise the "true" second preferences $\boldsymbol{d}_{\boldsymbol{t}}^{\boldsymbol{r,p2}}$ are implemented.

3 Simulation Experiments and Parameter Settings

In the simulation experiments (see Table 1), after a performance landscape is generated, the systems are placed randomly in the fitness landscape and observed over 250 periods while searching for superior solutions with respect to overall performance V (Eq. 2). The experiments introduced are intended to provide some evidence on the effects of erroneous communication occurring within the coordination of plans of distributed problem-solving agents and for different levels of coordination need. Therefore, the experiments distinguish for different levels of communication errors E^C and for different levels of complexity of the underlying search problem as captured by parameter K, and K^* respectively. Parameter K^* captures the level of cross-sub-problem interactions and with that, in fact, it reflects the need for coordination among the agents: If, for example, K^* equals zero, like in the block-diagonal structure, no interactions between sub-problems exist which could mutually affect the partial performance achieved by solutions for other sub-problems and, thus, no coordination is required.

Table 1. Parameter Settings

Parameter	Values / Types
observation period	$T = 250$
number of choices	$N = 12$
number of search agents	$M = 4$ with agent 1: $d^1 = (d_1, \cdots d_3)$, agent 2: $d^2 = (d_4, \cdots d_6)$, agent 3: $d^3 = (d_7, \cdots d_9)$, agent 4: $d^4 = (d_{10}, \cdots d_{12})$
interaction structures	block-diagonal: $K = 2; K^* = 0$; intermediate low: $K = 3; K^* = 1$; intermediate medium: $K = 7; K^* = 5$; full interdependent: $K = 11; K^* = 9$
coordination mode	sequential; horizontal veto
level of communication error	$E^C \in \{0, 1, 2, 3, 4\}$

In order to capture the complexity of the underlying $N = 12$-dimensional search problem and, in particular, the need for coordination across the $M = 4$ agents' sub-problems of equal size, simulations for four different interaction structures (see Table 1) are introduced: in the *block-diagonal* structure the overall search problem can be segmented into four disjoint parts with maximal intense intra-sub-problem interactions but no cross-sub-problem interactions [10]; hence, the level of cross-sub-problem interactions K^* is 0 and no coordination across agents is required. On the contrary, in the *full interdependent* case ($K = 11$, $K^* = 9$) all single options d_i affect the performance contributions of all other choices and the coordination need is maximal. Moreover, two intermediate levels of complexity are simulated, one with low ($K^* = 1$) and one with medium level ($K^* = 5$) of cross-agent interactions.

4 Results

In order to get some evidence on the robustness of the two coordination modes under investigation against communication errors the final performance, i.e., $V_{t=250}$, achieved in the end of the observation period is employed as a metric for the system's coherence. With respect to the robustness against erroneous communication the difference of the final performance obtained for an error level $E^C > 0$ against the final performance obtained for $E^C = 0$ is of particular interest. Figure 1 plots for each interaction structure and for each level of communication error under investigation this difference of final performances. Moreover, it is of interest whether the relative advantages of the two coordination modes against each other are affected by the level of erroneous communication. Figure 2 depicts the difference between the final performance achieved on average with the sequential mode and that reached employing the horizontal veto mode.

The results broadly, though not universally, support intuition that communication errors occurring in the course of coordination reduce the overall systems' performance. Moreover, the results suggest that with increasing coordination need among agents, as captured by K^*, the performance losses obtained with erroneous communication increase too. However, some aspects deserve closer attention:

First, as can be seen in Figure 1.b, erroneous communication apparently does not necessarily lead to performance losses: with low coordination need (i.e., K^*=1) communication errors appear to be beneficial in the horizontal veto mode. An analysis more into detail reveals that this reasonably is due to the effects of errors on the diversity of search: The horizontal veto mode is particularly prone to inertia (i.e., mutual blockading) since this coordination mode effectively prevents false positive decisions from the distributed problem solvers' perspective. However, the downside is that this may preclude that options which would increase the *overall* performance are chosen. With communication errors there is the chance that the "parochial" rejection of overall beneficial options is reduced and, thus, the solution space is more broadly explored. In particular, a more detailed analysis reveals that for E^C=0 in nearly 0.7 percent of the 250 periods a new configuration $\boldsymbol{d_t}$ is implemented; with E^C=1 this ratio is raised to 2.1 percent of periods, though about half of these alterations is in favor of false positive movements. [1] However, these findings correspond to results of prior research indicating that errors (in the assessment of options) could be beneficial [12].

Second, the results suggest that the performance losses compared to perfect communication remarkably depend on the coordination mode employed: In the sequential mode, the maximal loss is 37 points of percentage which occurs in case of full interdependence; with the horizontal veto mode at worst, 12 points of percentage are lost.

[1] Simulation experiments with an observation period of $T = 500$ yield similar results: Final performance achieved with E^C¿0 goes about 5 points of percentage beyond that level obtained with perfect communication; in 0.33 percent of the 500 periods with E^C=0 new solutions are implemented; for E^C¿0 this ratio goes up to 1.6 percent.

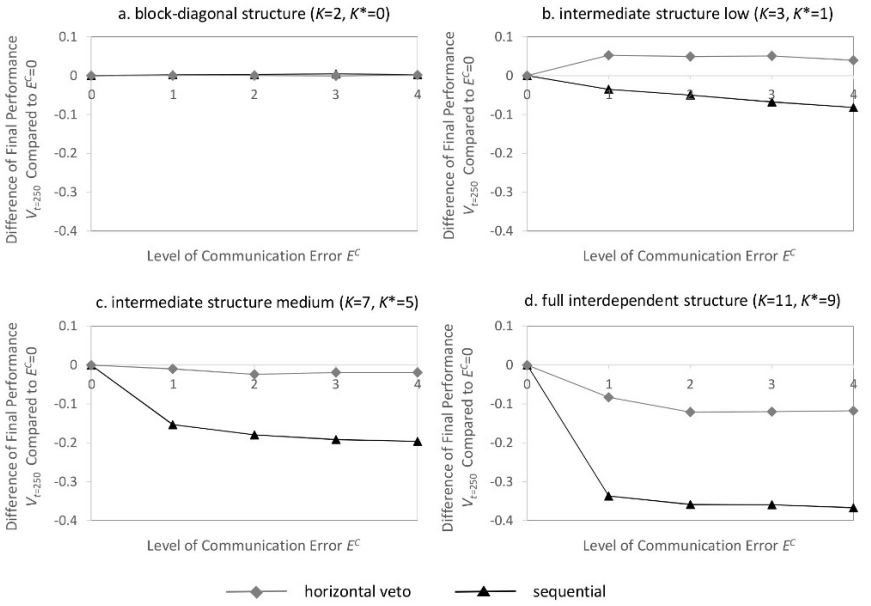


Fig. 1. Differences of final performance (left) and of frequency of global maximum found (right) achieved with communication error of level E^C compared to "no error" (E^C=0). Each mark represents the average of 2,500 adaptive walks, i.e., 250 distinct fitness landscapes with 10 adaptive walks on each. For parameter settings see Table 1. The 99.9 confidence intervals of averaged final performances $V_{t=250}$ obtained with the sequential (horizontal veto) mode of coordination show the following ranges: block-diagonal: ±0.0009 to ±0.0014 (±0.0011 to ±0.0013); intermediate low: ±0.0034 to ±0.0058 (±0.0032 to ±0.0048); intermediate medium: ±0.0072 to ±0.0074 (±0.0056 to ±0.0060); full interdependent: ±0.0046 to ±0.0075 (±0.0074 to ±0.0110)

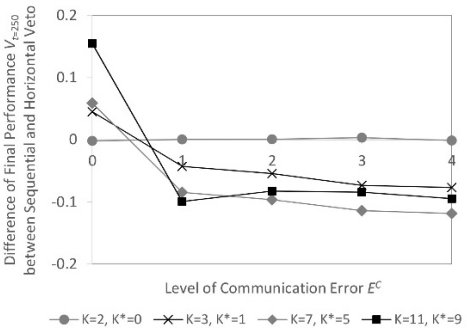


Fig. 2. Differences of final performance achieved with sequential and horizontal veto mode for different interaction structures. Each mark represents the difference between the averages of 2,500 adaptive walks, i.e., 250 distinct fitness landscapes with 10 adaptive walks on each. For parameter settings see Table 1

Third, the sequential mode which is ahead of the veto mode (Fig. 2) when communication works perfectly (e.g., for maximal coordination need, $K^* = 9$, with about 15.5 points of percentage) performs by 4 to 12 points of percentage worse than the horizontal veto mode in case of erroneous communication. In sum, the sequential mode appears to be rather sensitive towards communication errors and, in particular, even more sensitive than the horizontal veto mode.

Hence, an interesting question is what might cause the sequential mode's low robustness against imperfect communication. Reasonably, this is due to the propagation of errors incorporated in this mode. For example, an error occurring in the communication of agent 1's plan is propagated to all subsequent agents; not only does agent 2 adjust its partial plan to the distorted plan of agent 1; next, agent 3 takes the imperfectly communicated plan of agent 1 and the erroneously adjusted and imperfectly communicated plan of agent 2 into consideration and so on. Obviously, this propagation of errors is irrelevant if no interdependencies among agents exist (i.e., for $K^* = 0$) and becomes rather detrimental for maximal complexity (i.e., $K^* = 9$).

5 Conclusion

The major finding of this study is that the effects of erroneous communication subtly differ with the level of interdependencies between the sub-problems of a distributed problem-solving system and the coordination mechanisms employed. In particular, the results suggest that sequential planning, which provides high levels of performance when communication works perfectly, is particularly prone to imperfect communication. In contrast, the horizontal veto mode seems to be less sensitive to erroneous communication or, for levels of coordination need, even may benefit from communications errors.

The study presented in this paper calls for future research efforts. As such, this experimental study awaits its transfer to more practical applications in real world problems. This will reasonably also include to increase the number of agents and decisions and the organizational order among them. This would also include to study further modes of coordination including more centralized modes - given that some level of centralization of decision-making is feasible. Moreover, the interaction of erroneous communication should be studied in conjunction with further informational imperfections. In particular, in this study the agents are assumed to be capable of perfectly assessing the performance contributions of options, and, obviously, this is not universally the case.

References

1. Altenberg, L., Section B2.7.2, NK Fitness Landscapes. In: Bäck, T., Fogel, D. and Michalewicz, Z. (eds.): Handbook of Evolutionary Computation, Oxford University Press, Oxford, B2.7:5–B2.7:10, (1997).
2. Bond, A. H., Gasser, L.: Chapter 1 Orientation. In: Bond, A. H., Gasser, L. (eds.): Readings in Distributed Artificial Intelligence, Morgan Kauffmann, San Mateo, 156 (1988).
3. Garcia, E., Cao, Y., Yu, H., Antsaklis, P., Casbeer, D.: Decentralised event-triggered cooperative control with limited communication. International Journal of Control 86 (9):1479-1488 (2013).
4. Kauffman S.A., Levin S.: Towards a general theory of adaptive walks on rugged landscapes. J Theor Biol 128:11–45 (1993).
5. Kauffman, S.A.: The origins of order: Self-organization and selection in evolution. Oxford University Press, Oxford (1993).
6. Li, R., Emmerich, M. M., Eggermont, J., Bovenkamp, E. P., Bck, T., Dijkstra, J., Reiber, J. C.: Mixed-Integer NK Landscapes. Parallel Problem Solving from Nature IX: 4193:42–51, Springer, Berlin (2006).
7. Malone, T.W., Crowston, K.: The interdisciplinary study of coordination. ACM Comput Surv 26:87-119 (1994).
8. von Martial, F.: Coordinating Plans of Autonomous Agents, Lecture Notes in Artificial Intelligence, vol. 610, Springer: Berlin/Heidelberg/New York (1992).
9. Persis, C.D., Frasca, P.: Robust Self-Triggered Coordination With Ternary Controllers. IEEE Transactions on Automatic Control 58 (12):3024-3038 (2013).
10. Rivkin, R.W., Siggelkow, N.: Patterned interactions in complex systems: Implications for exploration. Manage Sci 53:1068–1085 (2007).
11. Tonino, H., Bos, A., de Weerdt, M., Witteveen: Plan Coordination by revision in collective agent based sysetms. Artificial Intelligence 142: 121–145 (2002).
12. Wall, F.: The (Beneficial) Role of Informational Imperfections in Enhancing Organisational Performance. Lecture Notes in Economics and Mathematical Systems 645, 115–126, Springer, Berlin (2010).
13. Wall, F.: Agent-based Modeling in Managerial Science: An Illustrative Survey and Study. Review of Managerial Science 10:135–193 (2016).
14. Wooldridge, M.: An Introduction to Multiagent Systems, 2nd Ed., Wiley, Chichester (2009).
15. Yongcan, C., Wenwu, Y., Wei, R., Guanrong, C.: An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination. IEEE Trans on Industrial Informatics 9:427–438 (2013)

A Sentiment Analysis Model to Analyze Students Reviews of Teacher Performance Using Support Vector Machines

Guadalupe Gutiérrez Esparza[1], Alejandro de-Luna[1], Alberto Ochoa Zezzatti[2], Alberto Hernandez[3], Julio Ponce[4], Marco Álvarez[1], Edgar Cossio[5] & Jose de Jesus Nava[6]

[1]Universidad Politécnica de Aguascalientes, Mexico.
[2]Universidad Autónoma de Ciudad Juárez, Mexico.
[3]Universidad Autónoma del Estado de Morelos, Mexico
[4]Universidad Autónoma de Aguascalientes, Mexico
[5]Universidad Enrique Diaz de Leon, Mexico
[6]Pinnacle Aerospace, Mexico

guadalupe.gutierrez@upa.edu.mx

Abstract. Teacher evaluation is considered an important process in higher education institutions to know about teacher performance and implement constructive strategies in order to benefit students in their education. The opinion of students has become one of the main factors to consider when evaluating teachers. In this study we present a Model called SocialMining using a corpus of real comments in Spanish about teacher performance assessment. We applied Support Vector Machines algorithm with three kernels: linear, radial and polynomial, to predict a classification of comments in positive, negative or neutral. We calculated sensibility, specificity and predictive values as evaluation measures. The results of this work may help other experiments to improve the classification process of comments and suggest teacher improvement courses for teachers.

Keywords. Support vector machines, teacher performance assessment, performance evaluation.

1 Introduction

In the teaching and learning process it is crucial to evaluate the teaching performance. This evaluation is one of the most complex processes in any university, since various factors and criteria should be met to be concentrated in order to provide a final assessment to the professional. Teacher evaluation can be performed by an observation guide or a rubric with different evaluation criteria. However, when teacher performance is evaluated by students, varied opinions are collected from the same established criteria. It is at this time when emerges the need to use sentiment analysis methods to the analysis these comments.

Sentiment Analysis [1] is an application of natural language processing, text mining and computational linguistics, to identify information from the text. Education is

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_19

one of the areas that currently have seen favorable sentiment analysis application, in order to improve attended sessions or distance education. In his research, Binali [2] ensures that students represents their emotions in comments, so it is a way to learn about various aspects of the student. Other research [3] use sentiment analysis on teacher's feedback from students enrolled in online courses in order to know their opinion and determine whether there is a connection between emotions and dropout rates. Student feedback on quality and standards of learning is considered as a strategy to improve the teaching process [3] and can be collected through a variety of social network, blogs and surveys.

In this paper, we presented a Model called SocialMining to support the Teacher Performance Assessment applying Support Vector Machines (SVMs). We selected SVMs as a classifier due to its high performance in classification applications. Further experiments with other machine learning algorithms will follow.

This paper is organized as follows. Section 2 presents related work. Section 3 shows the SocialMining Model architecture. Section 4 describes data and methods used, besides the experimental design. Section 5 includes the results obtained and a discussion. Finally, the conclusions of the work done, and the future work, are presented in Section 6.

2 Related work

The table 1 shows an overview of some related work. All these works have been successful in their different combinations of methods and algorithms. This table is not exhaustive.

Table 1. List of features to analize

Reference	Description	Algorithms used	Dataset used
[4]	Development of an application to know the student emotional state.	SVM, *Naïve Bayes, Complement Naïve Bayes.*	Students reviews at the University of Portsmouth
[5]	Development of an application, called SentBuk to retrieve identify users' sentiment polarity and emotional changes.	Lexicon based, machine-learning and hybrid approaches.	Around 1,000 comments in Spanish
[6]	Construction of a teaching evaluation sentiment lexicon, to analyzes the terms and phrases from student opinions	Lexicon in Thai, SVM, ID3 and Naïve Bayes.	Reviews by students at Loei Raja hat University.
[7]	Design of an experimental study to predict teacher performance.	Lexicon with 167 keywords positive and 108 keywords negative.	1,148 feedbacks bye students, obtained from RateMyProfessors.com

[8]	Proposed a method to detect the feeling of students on some topics and support the teacher to improve their teaching process.	Latent Dirichlet Allocation, SVM, Naïve Baye and, Maximum Entropy.	Movie reviews dataset [9] and comments by 75 students collected from Moodle.
[10]	Proposed a system to help the developer and educator to identify the most concentrated pages in E-learning portals.	Bayesian classification, Naïve Bayesian and SVM.	Use a collection of 100 users review from the website Functionspace.org.
[11]	Implement sentiment analysis in M-learning System, to know the user opinions of the M-learning system.	*Naïve Bayes*, KNN and *Random Forest.*	300 Reviews from www.market.android.com

From table 1 we can see that most of previous research has focused on particular aspects of Education. In this work we proposed a model to evaluate teacher performance consider Spanish reviews from students and applying machine learning algorithms to classify them as positive, negative and neutral. The results of this work may help to other experiments to improve the classification process of comments and suggest teacher improvement courses.

3 SocialMining model architecture

The SocialMining model is composed of three phases, where there a comments extraction process (teacher's feedback from students), a feature selection process and a comments classification into positive, negative and neutral, applying SVMs.

Phase 1. Comments extraction and cleaning process. At this phase we extracted teacher's feedback from students to generate a comments corpus. Then we do a labeling process to classify the comments into positive and negative considering a numeric range. The numeric range varied from: -2 to -0.2 is used for negative comments, -2 value express very negative comments. Values between +0.2 to +2 apply to positive comments, +2 is used as a positive comments. Likewise, those comments labeled with the number 1 are considered as neutral.

At cleaning process, the stop words and nouns that appear in most of the comments are deleted (e.g. teacher, university, class, subject, school and others). In addition, punctuation marks are removed and capitalized words are converted to lowercase.

Phase 2. Feature selection process. Once finished the cleaning process, we performed a feature selection process, removing repetitive terms and applying some functions to select the required terms or features, this process is like a filtering.

A feature in sentiment analysis is a term or phrase that helps to express an opinion positive or negative. There are several methods used in features selection, where some of them are based on the syntactic word position, based in information gain, using genetic algorithms and using trees like random forest importance variable.

At this phase it's necessary to know the importance of each feature, by their weight. So it's applied Term Frequency Inverse Document Frequency (TF-IDF).

Phase 3. Classification comments. At this phase the corpus of comments and features (matrixCF) is partitioned into two independent datasets. The first dataset is dedicated to training process (train) and is used in classification algorithms to find patterns or relationships among data; the second dataset is considered for the testing process (test) in order to adjust the model performance. In this work two thirds of the matrixCF are used for train dataset and one third for test dataset. Then the cross-validation method of k iterations is applied to control the tuning and training of SVMs algorithm. In this method matrixCF is divided into K subsets. One of the subsets is used as test data and the remaining (K-1) as training data. The cross-validation process is repeated for k iterations, with each of the possible subsets of test data, resulting in a confusion matrix with average values. Once the k iterations have completed cross validation accuracy is obtained. In this research, k is equal to 10.

The tuning process in SVMs allows adjusting the parameters of each kernel (linear, radial basis and polynomial). Then is performed a training process, through which identified whether the value of the parameters vary or remain constant.

Finally, the implementation of SVMs is performed presenting as a result the confusion matrix and accuracy values as well as the metric ROC.

4 Material and methods

4.1 Data

The dataset used in this work comprises 1040 comments in Spanish of three groups of systems engineering students at Polytechnic University of Aguascalientes. They evaluated 21 teachers in the first school grade (2016). For this study we considered only those comments free from noise or spam (characterized in this study as texts with strange characters, empty spaces no opinion or comments unrelated to teacher evaluation). In this work we identified a set of 99 features. An extract of the features are listed in table 2.

Table 2. List of features

Feature	Polarity feature	Feature	Polarity feature
Atenta	Positive	Debería	Negative
Agrada	Positive	Debe	Negative
Apoya	Positive	Impuntual	Negative
Aprender	Positive	Elitista	Negative
Bien	Positive	Impaciente	Negative
Bipolar	Positive	Problemático	Negative

4.2 Performance measures

We used typical performance measures in machine learning such as: accuracy, balanced accuracy, sensitivity, specificity, and metric ROC curve[12, 13].

4.3 Classifiers

SVMs is an algorithm introduced by Vapnik [14] for the classification of both linear and nonlinear data, it has been known for its quality in text classification [15]. There are kernels that can be used in SVMs, such as: linear, polynomial, radial basis function (RBF) and sigmoid. Each of these kernels has particular parameters and they must be tuned in order to achieve the best performance. In this work we selected the first three kernels to classify comments; this is mainly because of their good performance. The kernel used in this study, are: linear, radial basis and polynomial.

4.4 Experimental design

We created a dataset containing 1040 comments and 99 features associated with teacher performance assessment. We used train-test evaluation, two-thirds (2/3) for training and (1/3) one-third for testing, then performed 30 runs applying SVMs with polynomial, radial basis function (RBF) and linear kernel. For each run are computed performance measures. In each run we set a different seed to ensure different splits of training and testing sets, all kernels use the same seed at the same run.

Each kernel requires tuning different parameters (see table 3). A simple and effective method of tuning parameters of SVMs has been proposed by Hsu [16], the grid search. The C values used for the kernels, range from 0.1 to 2, the value of sigma (σ) varied from 0.01 to 2, the degree value parameter range from 2 to 10, and values between 0 and 1 are assigned for coef parameter. We performed 30 train-test runs using different seeds and calculated the accuracy and balanced accuracy of each run.

5 Discussion and Results

In this section, we present the experiments results with three kernels in SVMs. The first step is to determine the parameters of each kernel of SVMs, so we first load the data and create a partition of corpus of comments, then divided it into training and testing datasets, then use a traincontrol in R [17] to set the training method. We use the Hsu [16] methodology to specify the search space in each kernel parameter. ROC is the performance criterion used to select the optimal kernels parameters of SVMs.

Setting the seed to 1 in the process of optimization parameters we generated paired samples according to Hothorn [18] and compare models using a resampling technique. The table 3 shows the summary of resampling results using resamples in R [17], the performance metrics are: ROC, sensibility and specificity.

Table 3. Summary resampling results of parameters optimization

ROC						
Kernel	**Min.**	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**Max.**
Linear	0.7872	0.8315	0.8616	0.8579	0.8714	0.9080
Radial	0.7881	0.8218	0.8462	0.8462	0.8635	0.9176
Poly	0.8350	0.8608	0.8773	0.8755	0.8918	0.9321
Sensibility						
Kernel	**Min.**	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**Max.**
Linear	0.5857	0.6571	0.7	0.7006	0.7571	0.8143
Radial	0.6571	0.7286	0.7571	0.7663	0.8	0.9
Poly	0.3429	0.7	0.7571	0.7131	0.7714	0.8143
Specificity						
Kernel	**Min.**	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**Max.**
Linear	0.7971	0.8696	0.8971	0.889	0.9118	0.9565
Radial	0.6812	0.7681	0.7971	0.7925	0.8261	0.8696
Poly	0.8088	0.8529	0.8841	0.8849	0.913	0.9855

Once obtained optimized parameters for each kernel, the execution of each SVMs model is performed. The table 4 shows the average results of each kernel of SVMs across 30 runs. Also the standard deviation of each metric is presented.

Table 4. Average results across 30 runs in three kernels of SVM

	Accuracy	**balanced Accuracy**	**Sensitivity**	**Specificity**
SVM Linear	**0.8038**	**0.8149**	**0.8936**	**0.7160**
	0.0153	0.0146	0.0277	0.0364
SVM Radial	**0.7850**	**0.7893**	**0.8242**	**0.7467**
	0.0190	0.0159	0.0422	0.0561
SVM Poly	**0.6779**	**0.7014**	**0.8661**	**0.4941**
	0.0363	0.1336	0.1649	0.1009

The linear kernel obtained a balanced accuracy above 0.80. Values obtained in Sensitivity were much higher than those obtained in specificity in all kernels. The kernel polynomial (SVM Poly) had the lowest performance in all metrics except in sensitivity. The three kernels resulted more sensitive than specific.

6 Conclusions

In computer science its attractive the use of this type of machine learnings models to automate processes, save time and contribute to decision making. The SocialMining model supports the analysis of the behavior from unstructured data provided by students. The sentiment analysis is based on the analysis of texts and the SocialMining

can provide a feasible solution to the problem of analysis of teacher evaluation comments. Further experiments will be conducted in this ongoing research project.

It is important to point out that is necessary reduce the number of features through a depth analysis to identify the most relevant features of teacher performance assessment, in order to improve the results of comments classification process. Also we considered important have a corpus of comments balanced (positive and negative comments in equal quantity) for testing and training process.

In addition to conduct a deeper analysis for relevant features selection, we considered necessary implement other machine learning algorithms in order to measure the performance of each algorithm in the classification of comments and select the optimal with high accuracy results.

Based on the adequate results that have been obtained Model SocialMining applying Naïve Bayes and a corpus of subjectivity [19], it is considered that with the implementation of other algorithms of machine learning known for their good performance in classification process, the model proposed will support teachers to improve their classes. Another contribution SocialMining Model is that it could support the department of teacher improvement, to create and suggest different courses to teacher training skills, according to the possible opportunities that are detected in the comments of students.

References

1. Bing, L., *Sentiment Analysis and Opinion Mining (Synthesis Lectures on Human Language Technologies).* Morgan & Claypool Publishers., 2012.
2. Binali, H.H., Chen Wu, and Vidyasagar Potdar., *A new significant area: Emotion detection in E-learning using opinion mining techniques.* Digital Ecosystems and Technologies, 2009. DEST'09. 3rd IEEE International Conference on. IEEE,, 2009.
3. Wen, M., Yang, D., & Penstein Rosé, C., *Sentiment Analysis in MOOC Discussion Forums: What does it tell us?* . Proceedings of Educational Data Mining, 1 - 8. (http://goo.gl/fViyBH), 2014.
4. Altrabsheh, N., Cocea, M., & Fallahkhair, S., *Learning Sentiment from Students' Feedback for Real-Time Interventions in Classrooms.* Adaptive and Intelligent Systems. Volume 8779 of the series Lecture Notes in Computer Science, 40-49. doi: http://dx.doi.org/10.1007/978-3-319-11298-5_5, 2014.
5. Ortigosa, A., Martín, J. & Carro, R., *Sentiment analysis in Facebook and its application to e-learning.* Computers in Human Behavior. Volume 31, 527 - 541, http://dx.doi.org/10.1016/j.chb.2013.05.024, 2014.
6. Pong-inwong, C.R., W., *TeachingSenti-Lexicon for Automated Sentiment Polarity Definition in Teaching Evaluation.* In Semantics, Knowledge and Grids (SKG), 10th International Conference, 84 – 91, Beijing: IEEE., 2014.
7. Kaewyong, P., Sukprasert, A., Salim, N., and Phang, A., *The possibility of students' comments automatic interpret using lexicon based sentiment*

analysis to teacher evaluation. Conference: The 3rd International Conference on Artificial Intelligence and Computer Science 2015, At Penang,MALAYSIA, 2015.

8. F., C., M., de Santo and L., Greco, *SAFE: A Sentiment Analysis Framework for E-Learning.* International Journal of Emerging Technologies in Learning, Vol. 9, Issue 6, pp. 37, 2014.
9. Pang, B., & Lee, L., *Thumbs up? Sentiment Classification using Machine Learning Techniques.* EMNLP '02 Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10. Pages 79-86. doi>10.3115/1118693.1118704, 2002.
10. Bharathisindhu, P., & Brunda, S., *Identifying e-learner's opinion using automated sentiment analysis in e-learning.* IJRET: International Journal of Research in Engineering and Technology. Volume: 03 Issue: 01. doi: http://dx.doi.org/10.15623/ijret.2014.0319086, 2014.
11. Kirubakaran, A.J.a.E., *M-Learning Sentiment Analysis with Data Mining Techiniques.* International Journal of Computer Science and Telecommunications. Trichy-India., 2012.
12. I.H. Witten, E.F., and M. A. Hall., *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, third edition., 2011.
13. J. Han, M.K., and J. Pei, *Data Mining: concepts and techniques.* Morgan Kaufmann, San Francisco, USA., 2012.
14. Vapnick, V., *Statistical Learning Theory.* Wiley, New York, 1998.
15. Esuli, A.S., F., *SENTIWORDNET: a publicly available lexical resource for opinion mining.* . In Proceedings of the 5th Conference on Languaje Resources and Evaluation (LREC 2006). pages 417-422, Genoa, Italy., 2006.
16. C.C. Hsu, C.W.C.a.C.J.L., *A practical guide to support vector classification.* Technical report, Department of Computer Science, National Taiwan University, 2003.
17. R-Core-Team, *R: A language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria. (http://goo.gl/e40yiU), 2013.
18. Hothorn, T., Leisch, F., Hornik, K. and Zeileis A., *The design and analysis of benchmark experiments.* Journal of Computational and Graphical Statistics. Vol 14 (3) pp. 675-699., 2005.
19. Gutiérrez, G., Padilla, A., Canul-Reich, J., De-Luna, A. and Ponce, J., *Proposal of a Sentiment Analysis Model in Tweets for Improvement of the Teaching - Learning Process in the Classroom Using a Corpus of Subjectivity.* International Journal of Combinatorial Optimization Problems and Informatics, Vol. 7, No. 2, May-Aug 2016, pp. 22-34., 2016.

Proposal of Wearable Sensor-Based System for Foot Temperature Monitoring

J. Bullón Pérez[1], A. Hernández Encinas[2], J. Martín-Vaquero[2], A. Queiruga-Dios[2], A. Martínez Nova[3], and J. Torreblanca González[4]

[1] Department of Chemical and Textile Engineering, University of Salamanca, Spain
perbu@usal.es
[2] Department of Applied Mathematics, University of Salamanca, Spain
{ascen, jesmarva, queirugadios}@usal.es
[3] Department of Nursing, University of Extremadura, Spain
podoalf@unex.es
[4] Department of Applied Physics, University of Salamanca, Spain
torre@usal.es

Abstract. The design and development of wearable biosensor systems for health monitoring has garnered lots of attention in the scientific community and the industry during the last years. The Internet of Things (IoT) enables added value services by connecting smart objects in a secure way in different applications, such as transport, health, building, energy, ecology, or industry, through multiscale integration. This paper presents a preliminary study on the design of a smart sock for dia-betic patients. This smart sock will allow monitoring diabetic patient's health condition and temperature levels using foot temperature sensors. Sensed vital signs will be transmitted to a dedicated transmitter/receiver pair, such as a PC, PDA, or a mobile phone. It would make possible for patients to have real time information about their health condition, including foot temperature levels and therefore managing their diet and/or medication. As the market's demand for medical information increases, this smart sock will provide a significant part of the answer for patients.

Keywords: Health Care, Internet of Things, Smart Textile Systems, Wearable Textile.

1 Introduction

According to the International Diabetes Federation Atlas [9], by 2035 the diabetes mellitus will rise to almost 600 million, and around 80% of these people will live in developing countries. One of the problems suffered by diabetes patients is related to several complications in their lower limbs, more specifically in their feet. The severity of foot injuries is different depending on the socio-economic conditions of the patients, as the type of footwear and the general body care is not the same from region to region. In developed countries, foot ulcers has a yearly incidence of around 2-4% , but the percentage is higher in developing countries [5]. The foot ulcers are a result of peripheral sensory neuropathies,

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_20

minor foot trauma, foot deformities related to motor neuropathies, or peripheral artery disease. The consequence of a skin wound is that it become susceptible to infection, which means an urgent medical problem. Only two-thirds of foot ulcers have the appropriate treatment [13] and up to 28% result in lower extremity amputation. Every year, more than 1 million people with diabetes lose at least a part of their leg as a consequence of the complications of diabetes. Foot problems in diabetic patients represent a personal tragedy, and also affect patients's family. Moreover, this healthy problems suppose a substantial financial burden on healthcare systems and society in general. In low-income countries the cost of treating a complex diabetic foot ulcer can be equivalent to 5.7 years of annual income, potentially resulting in financial ruin for these patients and their families [6].

To avoid the high mortality rate for amputees (as a result of the diabetic desease) it would be necessary to have a diagnostic tool or a warning device for those patients. Patients suffering this metabolic disease lose the protective sensation caused by peripheral neuropathies. The development of such a warning device could allow patients the daily use, so it would provide them with feedback about when they are at risk.

Wearable health-monitoring systems have drawn a lot of attention from the research community and the industry during the last decade, as it is pointed out by the numerous and yearly increasing corresponding research and development efforts ([10], [14]). Wearable systems for health monitoring may comprise various types of miniature sensors, as wearable or even implantable devices. These biosensors are capable of measuring significant physiological parameters like heart rate, blood pressure, body and skin temperature, oxygen saturation, respiration rate, electrocardiogram, etc. The obtained measurements are communicated either via a wireless or a wired link to a central node, such as a smart phone or a microcontroller board, which may then in turn display the according information on a user interface or transmit the aggregated vital signs to a medical center. A wearable medical system may encompass a wide variety of components: sensors, wearable materials, smart textiles, actuators, power supplies, wireless communication modules and links, control and processing units, interface for the user, software, and advanced algorithms for data extracting and decision making [18].

Faced with this situation, we consider the possibility of using smart textiles (also known as electronic or e-textiles), i.e. "textiles that can detect and react to stimuli and conditions of the environment, as well as mechanical, thermal, chemical, electrical or magnetic stimuli" [7]. Therefore, it is the physical integration of an intelligent system with a textile substrate, in our case a system to monitor physiological signals.

2 The Internet of Things for Health Care

The IoT network for health care facilitates the transmission and reception of medical data, and enables the use of healthcare-tailored communications. The

architecture of those networks includes an outline for the specification of their physical elements, functional organization, and its working principles and techniques. Many studies [4], [8], [12], [20] have justified that the IPv6-based 6LoWPAN is the basis of the IoT healthcare networks. A service platform framework focusing on users' health information is presented in [23]. This framework shows a systematic hierarchical model of how caregivers can access various databases from the application layer with the help of a support layer.

IoT-based healthcare systems can be used in a wide range of fields, including care for pediatric and elderly patients, the supervision of chronic diseases, and the management of private health and fitness, among others. Those systems could be classified according two aspects: services (community healthcare monitoring, children health information, wearable device access, etc.), and applications (blood pressure monitoring, body temperature monitoring, rehabilitation system, healthcare solutions using smartphones, etc.) [11].

The IoT is growing rapidly. Therefore, the IoT healthcare domain may be the target of attackers. To facilitate the full adoption of the IoT in the healthcare domain, it is critical to identify and analyze distinct features of IoT security and privacy, including security requirements, vulnerabilities, threat models, and countermeasures, from the healthcare perspective.

The challenges for secure IoT healthcare services include: computational, memory and energy limitations; scalability; communications media; multiplicity devices; a dynamic network topology; a multi–protocol network; dynamic security updates and tamper resistant packages, see Figure 1.

3 Proposal of Wearable Sensor-Based System

A major challenge in the development of a wearable sensor-based system for foot temperature monitoring is to find the characteristics of diabetic feet that will be the indicators of the condition of the patient's feet. Commonly, pressure has been and remains to be an important indicator of ulceration ([15], [16], [17], [22]). The temperature is very closely associated with pressure, and it is a much more reliable parameter to measure. This is based, in part, on the available hardware. Thermistors that are highly accurate and also very small are readily available. However, less attention has been paid to plantar temperature as an indicator of ulcer risk as compared to pressure ([1], [2], [3]). Sadly, many patients go to the emergency department after reaching a point of no return in regards to their feet. Currently, there is no device available that can provide a portable and continuous monitoring and assessment of the condition of the feet in diabetic patients.

Technology has also entered the world of fashion. Fabrics and smart garments are the new –though not as popular– fashion trends in recent times. From clothing made of fibers that maintain blood pressure, to socks that delay the appearance of fungi, there are some of the latest discoveries in the development of fashion. There exists a trend directly linked to the current social concern of "feeling good", coupled with the demands of quality demanded by the con-

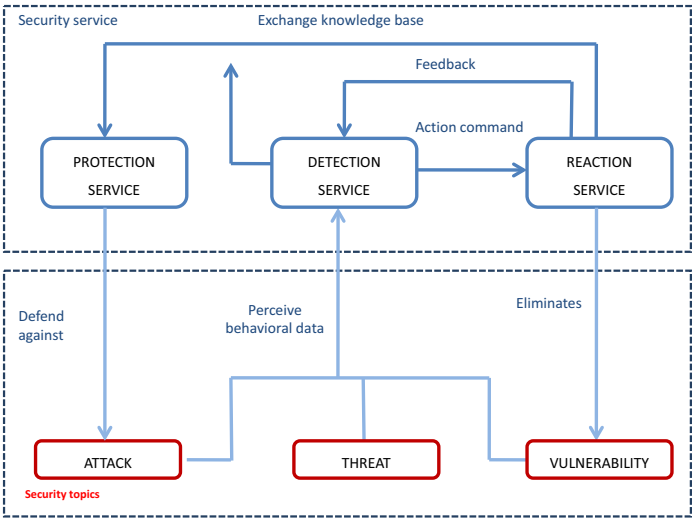


Fig. 1. Intelligent security model.

sumer. In this new fabrication of textiles, it is not only clothing that has healthy and often ecological characteristics, there is also a line of researchers who have developed smart textiles that change color, emit sound, etc.

In addition to the purpose of the design, smart textiles can also play a role in the health area. Such is the case of fabrics with metallic threads that can give information about the body, clothes that can measure pulse and temperature, or heart rate and breathing. The manufacturer of polyester Advansa has developed a model that incorporates the assets of ThermoCool to regulate the temperature. It was made together with the Chinese firm CAML (China Access Marketing Ltd.), specialized in socks for the practice of the sport (`http://www.camlhk.com/`). CAML is the first Chinese socks manufacturer to apply ThermoCool to their products. The company, known for its line of hosiery for all types of winter sports and, especially, for the practice of trekking, has launched a new special collection for cyclists and runners with ThermoCool.

Some researchers from the University of Manchester have developed a system to take measures of temperature in diabetic feet to study the etiology of diabetic foot ulcerations [19]. In our study we are going to place several sensors in a sock in order to take measurements of temperature in different places of the feet.

The proposed IoT-based device includes a system for monitoring the foot's temperature. It is based on the following main components (see Fig. 2):

– Textile-based sensors in a smart sock, that will receive the data (foot temperature).

– A central unit (base station), connected to the smart sock.
– An external device that receives the wireless information sent by the central unit and estimates the dataset.

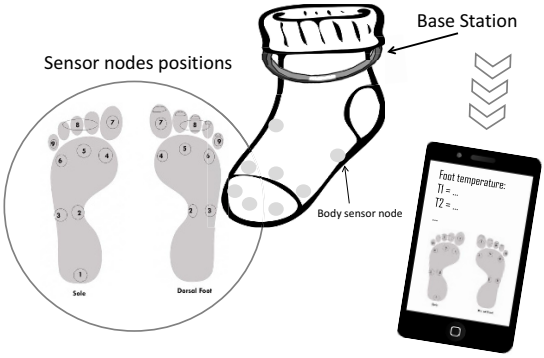


Fig. 2. Scheme of the proposed wearable to measure temperature data.

The project of a smart sock for diabetic patients started with the search for the published work, using as key words the temperature methods and the diabetic foot. Moreover, in order to find the relevant devices for collecting foot temperature data, the search had been widened to the European patents online platform. This research was based on the instruments for the evaluation of diabetic foot temperature, as well as its origin, technology base, and applications. In addition, the search strategy was to find patents and patent applications, and also look for updates and improvements. In order to find inventions and non–patentable equipments, already available in the market, the search was conducted through the Internet, establishing the following keywords in a search engine: diabetic foot, foot temperature, monitoring temperature, evaluation of temperature, and technology.

A smart insole for diabetic patients [21] is a temperature based smart insole capable of continuously or intermittently measuring the foot temperature of the patient at one or more locations of the foot while the insole is worn. This device provides the feedback to the patient, warning about danger based on user's plantar temperatures. Furthermore, it frees the patient from the clinical setting and increases patient's confidence to be mobile, while at the same time allows the patient self–monitoring their feet. The smart insole monitors foot temperature with the help of several temperature sensors. In addition to this, the device needs an algorithm to compare the data from the temperature sensors with a individual's signature profile, and provides a feedback value. Some additional

components are needed for communicating the feedback value. Another example of this smart system also incudes a plurality of temperature sensors that generate a signal. The signal is collected using a circuit electrically connected to the temperature sensors. A software program receives the transmitted signal and compares the transmitted signal to a signature profile to generate a feedback signal that is transmitted to a final device.

The medical stocking for temperature detection proposed in [24] is provided with a temperature sensor and an indicator connected to the sensor for providing a signal indicative or leg temperature. The stocking is additionally provided with at least one strip of soft but substantially rigid material for inhibiting a rolling down of the stocking. The strip extends longitudinally along the stocking from an edge of the stocking's mouth. In addition, a pressure application component is provided on the stocking for automatically applying periodic compressive pressure to the person's leg.

To be more precise, the wearable sensor-based system proposed in this study, for foot temperature monitoring, will include:

- Temperature sensors to collect the temperature data of the sock.
- The foot temperature profile of the individual (based on the temperature readings from strategic locations of the foot).
- Based on a fuzzy inference engine, the sock makes a decision on whether to alert the patient.
- The alert signal is transmitted in various ways, as visual, tactile or audible indication, and combinations thereof.
- The alert signal can be transmitted wirelessly to a pager-type receiver carried by the patient, a caregiver, or a physician's office, for example.
- The system can be powered including one or more batteries which may be rechargeable or non-rechargeable.

Although the wearable had not yet been developed, temperature measurements were performed with different sensors, including the LM35 and NTC thermistors. The first measurements have been taken with LM35 sensors connected to a Microchip PIC series microcontroller. The initial sensor with which we are working is the LM35 for its easy implementation and its response in circuit.

The LM35 is a temperature sensor, which is calibrated to an accuracy of 1°C and is capable of measuring between -55°C and 150°C. It has 3 connection pins. One is for power, one for mass (0 Volts) and one for the data output signal. It is very simple to use. The extreme pins of LM35 are for power, while the center pin provides the measurement at a voltage reference, at the rate of 10mV / °C. The microcontroller with which we are going to perform the data acquisition is a PIC16F877A which has 40 terminals. Of all of them we are going to use the eight destined to the analog-digital converter, an analog signal for its inputs will be converted into a digital signal, which we will proceed to try to obtain a result, which will be the temperature of the sensor in question.

4 Conclusions and further work

Textile-based sensors are always made of textiles and define themselves through their textile structure. We are developing a prototype as a smart sock that will contain several sensors to measure real temperature data. These sensors will be placed into the sock plants.

The proposed device is a wearable sensor-based system for foot temperature monitoring capable of continuously or intermittently measuring the foot temperature of the diabetic patients at several locations of the feet.

The device will provide feedback to the patients, alerting about the risk based on his/her feet temperatures.

The benefits of this smart device will include:

- To free the diabetic patients from the clinical setting and increase patient's confidence to be mobile, allowing to self-monitor their feet.
- Diabetic patients with peripheral neuropathy could wear the socks and be automatically alerted when their foot temperature has exceeded a determined profile.
- Can be used to diagnose or monitor the foot diseases or disorders. Some foot diseases or disorders cause an increase in temperature at certain locations of the foot.

Acknowledgements

The authors acknowledge support from the Fundación Memoria D. Samuel Solórzano Barruso, through the grant FS/14-2016.

References

1. Armstrong, D.G., Lavery, L.A.: Monitoring neuropathic ulcer healing with infrared dermal thermometry. The Journal of foot and ankle surgery 35(4), 335–338 (1996)
2. Armstrong, D.G., Lavery, L.A., Liswood, P.J., Birke, J., et al.: Infrared dermal thermometry for the high-risk diabetic foot. Physical Therapy 77(2), 169 (1997)
3. Benbow, S.J., Chan, A.W., Bowsher, D.R., Williams, G., Macfarlane, I.A.: The prediction of diabetic neuropathic plantar foot ulceration by liquid–crystal contact thermography. Diabetes care 17(8), 835–839 (1994)
4. Bouaziz, M., Rachedi, A.: A survey on mobility management protocols in wireless sensor networks based on 6lowpan technology. Computer Communications 74, 3–15 (2016)
5. Boulton, A.J., Vileikyte, L., Ragnarson-Tennvall, G., Apelqvist, J.: The global burden of diabetic foot disease. The Lancet 366(9498), 1719–1724 (2005)
6. Cavanagh, P., Attinger, C., Abbas, Z., Bal, A., Rojas, N., Xu, Z.R.: Cost of treating diabetic foot ulcers in five different countries. Diabetes/metabolism research and reviews 28(S1), 107–111 (2012)
7. Cherenack, K., Zysset, C., Kinkeldei, T., Münzenrieder, N., Tröster, G.: Woven electronic fibers with sensing and display functions for smart textiles. Advanced materials 22(45), 5178–5182 (2010)

8. Doukas, C., Maglogiannis, I.: Bringing iot and cloud computing towards pervasive healthcare. In: Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on. pp. 922–926. IEEE (2012)
9. Federation, I.D.: IDF Diabetes Atlas. International Diabetes Federation, Bruselas (2013)
10. Gatzoulis, L., Iakovidis, I.: Wearable and portable ehealth systems. IEEE Engineering in Medicine and Biology Magazine 26(5), 51–56 (2007)
11. Islam, S.R., Kwak, D., Kabir, M.H., Hossain, M., Kwak, K.S.: The internet of things for health care: a comprehensive survey. IEEE Access 3, 678–708 (2015)
12. Jara, A.J., Zamora, M.A., Skarmeta, A.F.: Knowledge acquisition and management architecture for mobile and personal health environments based on the internet of things. In: 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications. pp. 1811–1818. IEEE (2012)
13. Jeffcoate, W.J., Chipchase, S.Y., Ince, P., Game, F.L.: Assessing the outcome of the management of diabetic foot ulcers using ulcer-related and person-related measures. Diabetes care 29(8), 1784–1787 (2006)
14. Lmberis, A., Dittmar, A.: Advanced wearable health systems and applications - research and development efforts in the european union. IEEE Engineering in Medicine and Biology Magazine 26(3), 29–33 (2007)
15. Manley, M., Darby, T.: Repetitive mechanical stress and denervation in plantar ulcer pathogenesis in rats. Archives of physical medicine and rehabilitation 61(4), 171–177 (1980)
16. Martínez-Nova, A., Cuevas-García, J.C., Pascual-Huerta, J., Sánchez-Rodríguez, R.: Biofoot® in-shoe system: Normal values and assessment of the reliability and repeatability. The Foot 17(4), 190–196 (2007)
17. Mueller, M.J., Smith, K.E., Commean, P.K., Robertson, D.D., Johnson, J.E.: Use of computed tomography and plantar pressure measurement for management of neuropathic ulcers in patients with diabetes. Physical therapy 79(3), 296 (1999)
18. Pantelopoulos, A., Bourbakis, N.G.: A survey on wearable sensor-based systems for health monitoring and prognosis. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 40(1), 1–12 (2010)
19. Reddy, P.N., Cooper, G., Weightman, A., Hodson-Tole, E., Reeves, N.: An in-shoe temperature measurement system for studying diabetic foot ulceration etiology: preliminary results with healthy participants. Procedia CIRP 49, 153–156 (2016)
20. Shelby, Z., Bormann, C.: 6LoWPAN: The wireless embedded Internet, vol. 43. John Wiley & Sons (2011)
21. Shoureshi, A., Albert, A.: Smart insole for diabetic patients (May 2008), uSA Patent. US7716005
22. Stess, R.M., Jensen, S.R., Mirmiran, R.: The role of dynamic plantar pressures in diabetic foot ulcers. Diabetes care 20(5), 855–858 (1997)
23. Wang, W., Li, J., Wang, L., Zhao, W.: The internet of things for resident health information service platform research. In: IET International Conference on Communication Technology and Application (ICCTA 2011). pp. 631–635. IEEE (2011)
24. Wilk, P.J.: Medical stocking for temperature detection (Aug 1996), uSA Patent. US5546955

Modeling and checking robustness of communicating autonomous vehicles

Johan Arcile[1], Raymond Devillers[2], Hanna Klaudel[1],
Witold Klaudel[3], and Bożena Woźna-Szcześniak[4]

[1] IBISC, University of Evry, 23 bd de France, 91037 Evry, France,
{johan.arcile,hanna.klaudel}@ibisc.univ-evry.fr
[2] ULB, Bruxelles, Belgium, rdevil@ulb.ac.be
[3] Renault SA, 1 av. du Golf, 78084 Guyancourt, France, witold.klaudel@renault.com
[4] IMCS, Jan Długosz University in Częstochowa. Al. Armii Krajowej 13/15, 42-200
Częstochowa, Poland b.wozna@ajd.czest.pl

Abstract This paper presents a method for the validation of communicating autonomous vehicles (CAVs) systems. The approach focuses on the formal modeling of CAVs by means of timed automata, to allow the formal analysis through model-checkers of the vehicle behavior, including their fault tolerance due to various kinds of injected faults. We also present our case studies results, based on implementations of our CAVs' model in UPPAAL.

Keywords: Timed Automata, Model-Checking, Communicating Vehicles

1 Introduction

Autonomous vehicles are rational and sophisticated entities (agents) that, as the term already suggests, act autonomously (having different informations and/or diverging interests) across open and distributed environments. A *communicating* autonomous vehicles (CAVs) system is both a multi–agent system [10] and a real-time system [2]. More precisely, a CAV system is a network of vehicles that communicate to fulfill a mission as fast as possible while complying to the code rules and avoiding conflicts. Moreover, its correctness also depends on respecting a set of time constraints.

The CAVs systems (*e.g.* Google Car [5], VISLAB [9] or MB [6] prototypes), have the potential to substantially affect safety, congestion, energy consumption, and, ultimately, will likely reduce crashes and pollution. Thus, the reliability of CAVs systems is a key concern of policymakers [3]. It is therefore appealing to formally verify the core CAVs behaviors in order to be confident in the integration of autonomous vehicles into the road traffic.

Formal modeling and verification of CAVs systems require not only the definition of both the vehicle states and the road (the environment) but also a specification of interactions between vehicles and an expressive property language. The resulting model of a CAVs system should be accurate enough to capture spatial and time aspects of the original system and also should provide a formal basis for verification of properties like effectiveness of overtakings, the resulting arrival order, and the impossibility of

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_21

collisions (safety). The property language should be easy and appropriate for expressing the needed properties and applying automatic verification techniques for them.

Model checking [4] is a widely used automatic method for system verification. It provides algorithmic means for determining whether an abstract model – representing a hardware or software project (e.g., a CAVs system) – satisfies a formal specification expressed as a temporal logic formula. Moreover, if the property does not hold, the method identifies a counterexample execution that shows the source of the problem.

The main objective of this paper is to present our ongoing work on formal modeling and model checking of a communicating autonomous vehicles (CAVs) system, focusing in particular on the robustness of the decision policy. The modeling formalism that we use in our approach to specify the behavior of CAVs is a network of Timed Automata [2] with the synchronous semantics. We have chosen this formalism, because it is the most well-established model for the specification and verification of distributed real-time system design. Moreover, the automata formalism is a standard supported by several verification tools, e.g., the model checker UPPAAL [8].

The timed automata formalism allows us to create a clear and concise abstract model of the considered CAVs system. The formalism allows for assessing a vehicle decision policy robustness through a fault injection, and allows for applying model checking algorithms, in particular the algorithms designed for timed properties expressed in the temporal logic TCTL [1]. To the best of our knowledge, this kind of formal approach does not seem to have been exploited up to now, if we except attempts like [[7]] where very simple (untimed) automata are used.

2 Modeling

The systems we consider are composed of several lanes forming a portion of a motorway on which several vehicles can move in the same direction, each one realizing some goal (mission). Each vehicle is defined by constants such as its length or breaking capacities. The vehicles are provided with various perception sensors allowing to observe the behavior of their neighbors. They are also able to communicate informations, which are not always directly observable such as their intention of lane change, which may be available immediately or after some delay. A vehicle behavior consists in performing, at its own frequency: (i) a computation allowing to make a decision on the immediate action to be performed (*i.e.*, execution of the controller algorithm) followed by (ii) the communication of its intention (and possibly when the maneuver could start if not immediately) to all vehicles around (at least to those, which are capable to receive this information). Received information coming from the other vehicles is stored in the vehicle's database and used when running the controller algorithm. The controller algorithm computes predictions of trajectories of the neighbors allowing in turn to compute the instructions for itself in terms of acceleration or lane change, all this of course has to be done safely and obeying traffic laws.

On this basis, we build an abstract but realistic enough model in timed automata that may be analyzable with various temporal logic queries in a reasonable[5] time.

[5] a few minutes, or hours for very complex systems and queries (but recall this is the time needed to analyze models off-line, not the time for a vehicle to react on the road)

In our framework, we consider a road section composed of one or more unidirectional lanes the vehicles move on, and we store at any time the coordinates (position) of each vehicle on it. The current position (x,y) (as well as the initial one) is expressed as the distance from the beginning of the road section (x) and from the road border (y). Due to verification purposes requiring discrete values domains, these distances are expressed using discrete values, but precise enough to satisfactorily model the vehicle progression on the road. As a consequence, the position of a vehicle is a point on a two-dimensional orthogonal grid on which the longitudinal and lateral gaps between adjacent points may be different.

A vehicle's mission is defined as a sequence of positions the vehicle has to visit. More precisely, we consider a sequence of sets of positions to be reached, with the requirement to reach at least one position of each set. Possible maneuvers a vehicle may launch in order to fulfill its mission are: keeping/changing lane and accelerating/decelerating or keeping its speed.

Each vehicle state is described by a record containing its position (the vehicle's center), its current speed, its direction (to the left, to the right or no change), and its acceleration (taken from a given range of possibilities, negative, positive and null). Its behavior is modeled by a timed automaton in the form of a loop which sets at some frequency the vehicle's acceleration and current direction. These indications are the result of the vehicle's controller algorithm, which can be different for each vehicle.

The vehicles' movements are modeled by another timed automaton, the environment, which computes and updates at every sample period all vehicles speeds and positions, according to the indications given by the vehicle's controllers.

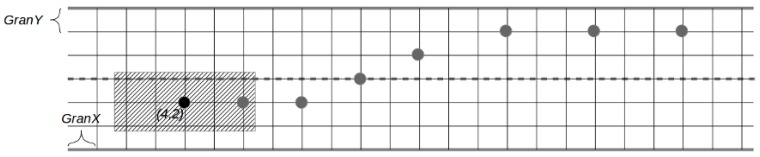


Figure 1. A two lanes road section discretized in abscissa with a granularity Gran_x and in ordinate with Gran_y. A vehicle (dashed rectangle) is centered in $(4,2)$ and its predicted timed trajectory $(6,2);(8,2);(10,3);(12,4);(15,5);(18,5);(21,5)$ for seven next time periods is shown by grey dots.

For our experiments, we developed a controller in a way that the goal of a vehicle is to reach positions of its mission as fast as possible while complying to the code rules and avoiding conflicts with other vehicles. To do so, the vehicle computes timed trajectories of its neighbors from their positions, speeds and intentions, and chooses its own trajectory regarding to their.

Vehicle's predicted trajectories are represented using timed abstractions. In order to optimize the state space, the trajectories are never stored but they are computed on demand from the information about vehicles, which has to be as compact as possible.

Thus, the timed trajectories are represented as sequences of positions the vehicle will reach at some dates up to some time horizon (typically 10 s), as shown in Fig. 1.

3 Discussion on modeling choices and parameters

Our objective is to obtain a realistic modeling of a communicating autonomous vehicles system that allows their model checking in a reasonable time. Since the available tools that are able to formally verify such systems can only handle integer data structures, we have to propose a satisfactory discretization for the various physical quantities describing the vehicle behaviors. Moreover, the complexity of the verification procedures rapidly increases with the range of these integer variables, while the accuracy of the models requests an adequate granularity.

Our discrete representation of a vehicle's state will thus be encoded with the following discrete variables: its (longitudinal and lateral) positions x and y, its forward speed v with the corresponding forward acceleration a, and the direction of the vehicle $D \in \{-1,0,1\}$, corresponding respectively to a lateral move to the right, no change and to the left. With the exception of D, we will denote by Gran_a, Gran_v, Gran_x, Gran_y the granularity of these quantities, *i.e.*, the gap between two consecutive values (this allows to normalize the data as integers), and by N (with the corresponding subscripts) the size of the needed data structure, called a *range*.

The updates of the forward speed and position values after a period are expressed by the usual formulas. In order to adapt the data structures to the verification constraints, we want to make the system parametric depending on the value of the time period, called the *sample*, *i.e.*, the period with which the system is observed. Thus, to describe the system let us consider the following parameters and constants:

- S is the sample period, expressed in seconds (written as s);
- L is the length and R is the width of the road segment, in meters (written as m);
- V_{min} is the min and V_{max} is the max value of (longitudinal) speed expressed in meters per second (written as m/s);
- A_{min} is the min and A_{max} is the max value of acceleration expressed in meters per second squared (written as m/s/s);
- Gran_a is the granularity of the acceleration expressed in meters per second squared;
- W is the maximal value of the lateral speed expressed in meters per second.

This yields the acceleration range $\mathsf{N}_a = 1 + (A_{max} - A_{min})/\mathsf{Gran}_a$; we assume that 0 is one of the possible accelerations, and that A_{max}/Gran_a as well as A_{min}/Gran_a are integers. The acceleration is then expressed as $a = A \cdot \mathsf{Gran}_a$, longitudinal speed as $v = V \cdot \mathsf{Gran}_v$, longitudinal position as $x = X \cdot \mathsf{Gran}_x$, and lateral position as $y = Y \cdot \mathsf{Gran}_y$, where A, V, X, Y are integers.

Now, we are able to compute the granularities and ranges for the longitudinal and lateral positions and speeds, as well as give their normalized updates:

- For longitudinal speed, the update after one sample is $v' = v + a \cdot \mathsf{S}$. For a given granularity Gran_v, this lead to $V' = V + (A \cdot \mathsf{S} \cdot \mathsf{Gran}_a)/\mathsf{Gran}_v$ (in normalized variables). We choose a granularity $\mathsf{Gran}_v = \mathsf{Gran}_a \cdot \mathsf{S}$ (without new losses) leading to a normalized speed update $V' = V + A$. The resulting range is $\mathsf{N}_v = 1 + (V_{max} - V_{min})/(\mathsf{Gran}_a \cdot \mathsf{S})$, using a ceiling function if the division does not provide an integer.

- For the longitudinal position, the update after one sample is $x' = x + v \cdot \mathsf{S} + a \cdot \mathsf{S}^2/2$. For a given granularity G_x this leads to $X' = X + ((V \cdot \mathsf{S} \cdot \mathsf{Gran}_v) + (A \cdot \mathsf{S}^2/2 \cdot \mathsf{Gran}_a))/G_x = X + (2V + A)(\mathsf{Gran}_a \cdot \mathsf{S}^2/2)/G_x$ (in normalized variables). In order to avoid losses, we should choose the granularity $G_x = \mathsf{Gran}_a \cdot \mathsf{S}^2/2$. However this granularity will usually be too small (for instance, if $\mathsf{Gran}_a = 1$ and $\mathsf{S} = 10^{-1}$, we get a granularity of 5 mm), leading to a huge range. In order to avoid a state space explosion, we shall thus approximate x with a precision of $\mathsf{Gran}_x = p \cdot G_x$, with an adequate parameter p. Hence, the normalized update of x becomes $X' = X + (2V + A)/p$ (rounded) and the corresponding range is $\mathsf{N}_x = \mathsf{L}/\mathsf{Gran}_x$. In order to choose a suitable p, we may observe that Gran_x is the maximal longitudinal loss of precision we may face during a sample. Hence, we may introduce the (normalized, maximal) loss of precision[6] during one second $\mathsf{Norm}_x = \mathsf{Gran}_x/\mathsf{S} = p \cdot \mathsf{Gran}_v/2$. Norm_x may be fixed independently from the constants of the program and we get the corresponding $\mathsf{Gran}_x = \mathsf{Norm}_x \cdot S$ and $p = 2 \cdot \mathsf{Norm}_x/\mathsf{Gran}_v$.
- For the lateral position update after one sample we have $y' = y + W \cdot \mathsf{S} \cdot D$, leading to a granularity $\mathsf{Gran}_y = W \cdot \mathsf{S}$ and the range $\mathsf{N}_y = \mathsf{R}/\mathsf{Gran}_y$. In the normalized variables, this becomes $Y' = Y + D$.

The size E of the state space per vehicle, due to the data structure, is obtained by multiplying all the above variable ranges, while it becomes E^n for n vehicles. Fortunately, the formal tools do not necessarily construct the whole state space to analyze such systems. Of course, the complexity of the verification procedures also depends on the number of clocks used in the model and on the number of time zones they generate, as well as on the degree of non determinism present in the specification and the difficulty to solve the queries.

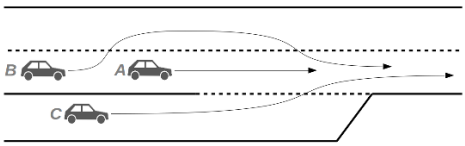
4 Experiments with the model

We illustrate our approach on a scenario implying three vehicles evolving on a three-lane highway section of $\mathsf{L} = 500$ m long and $\mathsf{R} = 10.5$ large. One of the lanes is a junction lane, which starts at the beginning of the section, joins the right lane after 200 m and ends 200 m later. The constraints defining the velocity of the vehicles are $V_{min} = 0$ m/s and $V_{max} = 40$ m/s (= 144 km/h); $A_{min} = -5$ m/s/s and $A_{max} = 3$ m/s/s, and $W = 1$ m/s. We fix $\mathsf{Gran}_a = 1$ m/s/s and the environment sample $\mathsf{S} = 0.1$ s (10 Hz). Such an S allows to monitor the system's behavior in a satisfactory way and such a Gran_a yields a sufficient number of acceleration choices for the vehicle's controllers.

With such parameters and the granularity guaranteeing no further loss of information, *i.e.*, $G_x = 0.005$ m, the complexity of the data structure per vehicle is of the order of 10^{11} and it becomes $(10^{11})^3$ for our 3 cars scenario, which is problematic for the verification tools. We then fix a reasonable normalized accuracy $\mathsf{Norm}_x = 1$ and get $p = 2 \cdot q/\mathsf{Gran}_v = 20$. Therefore, the granularity on x becomes $\mathsf{Gran}_x = p \cdot G_x = 0.1$ m and this approximation allows to divide the state space by $p^3 = 8000$ while not significantly impacting the behavior of the model.

We aim at studying the vehicle's controllers decisions (working at a period of 0.11 s) and their impact on each others. The vehicles A, B and C are initially positioned as

[6] Actually two times the loss of precision thanks to rounding.



Query\Variant	Base case	B-slowed	C-slowed	C-moved
Exists B overtakes A	T	F	T	T
Always B overtakes A	F	F	F	F
Exists C overtakes A	T	T	F	T
Always C overtakes A	T	T	F	F
Exists B overtakes C	F	F	T	T
Always B overtakes C	F	F	T	T

Figure 2. Top: initial situation of the Base Case (BC) with $x_A = 50$, $x_B = 0$, $x_C = 20$, $v_A = 20$, $v_B = 35$ and $v_C = 30$; B-slowed is BC with $v_B = 30$, C-slowed is BC with $v_C = 25$ and C-moved is BC with $x_C = 15$; bottom: queries and results.

shown in Figure 2 (top). They are assumed to have initially a null acceleration and their mission is to reach the right lane at the end of the road.

For a number of initial situations, we consider a series of properties – expressed as CTL formulas [4] – to check the effectiveness of overtakings, the resulting arrival order, and of course the possibility of collisions. The table in the bottom of Figure 2 gathers the results of the model checking for the base case defined above and a few of its variants.

For the Base Case, we can observe that it is possible that B overtakes A, which is slower, but this does not happen in every possible path. We can also observe that C always overtakes A but is not overtaken by B, meaning that C always arrives first. For B-slowed, the only difference with the Base Case is that B can no longer overtake A, independently of the order in which the vehicles take their decisions. For C-slowed, we can see that C now never overtakes A and is always overtaken by B, which means that C always lets the two other go before changing lane. For C-moved, we can see that the only difference with C-slowed is that C always waits for B to go first but then there are some paths where it goes before A and some where it doesn't. To explain this behavior, a hypothesis could be that C always waits for B, and when A is overtaken by B, C also overtakes A. We can check this with another query $(B \text{ overtakes } A) \rightarrow (C \text{ overtakes } A)$, which is indeed true. Finally, there is no path in any scenario that allows collision.

Despite the high number of states, UPPAAL handles most of the queries in less than 10 seconds on an office laptop. Unfortunately, in some cases such as for example C-slowed, some queries needed to explore a large part of the state space and took more than 4 minutes, meaning that the approximation on the position was necessary.

5 Experiments with the model after fault injection

The scenarios analyzed in section 4 were assumed to take place in a perfect environment, where each vehicle had a perfect knowledge of the other vehicles, thanks to

its sensors and the communication devices, with no loss of informations (but the basic discretization). It resulted in behaviors that were always safe (*i.e.*, the controllers always made safe choices). Here, we address the assessment of robustness of the system by injecting various kinds of faults. Concretely, we will degrade the environment by modifying the knowledge of some vehicles, representing either defective sensors or communication problems, such as for example a larger latency. We shall thus consider the same properties as before on the following degraded versions of the base case:

- Var-Null: the other vehicles' acceleration is interpreted as 0;
- V-Up: the other vehicles' speed is multiplied by $3/2$;
- V-Down: the other vehicles' speed is multiplied by $2/3$;
- X-Up: the other vehicles' longitudinal position is increased by 2 m;
- X-Down1: the other vehicles' longitudinal position is decreased by 5 m;
- X-Down2: the other vehicles' longitudinal position is decreased by 10 m.

Those faults impact the trajectory that each vehicle computes in order to represent the behavior of its neighbors. Table 1 gathers the model checking results for those versions.

Query\Variant	Var-Null	V-Up	V-Down	X-Up	X-Down1	X-Down2
Exists *B* overtakes *A*	F	T	F	F	T	T
Always *B* overtakes *A*	F	F	F	F	F	T
Exists *C* overtakes *A*	T	T	T	T	T	T
Always *C* overtakes *A*	T	T	T	T	T	T
Exists *B* overtakes *C*	F	T	F	F	F	F
Always *B* overtakes *C*	F	F	F	F	F	F
Exists collision	F	T	F	T	F	T

Table 1. Meaning of the queries with the corresponding results for each kind of fault.

For Var-Null, where each vehicle is unable to know the value of the other vehicles acceleration, and V-Down, where they use a speed value lower than the reality, we observe that B does not succeed in overtaking A anymore, but there is no collision. X-Up shows the same behavior for B, but a collision is then possible. The trace given by the model checker shows that B crashes with A while trying to overtake it. For V-Up, where the speed value of any other vehicle is increased, we observe that B sometimes overtakes C, and a collision is also possible. The trace given by the model checker shows that B crashes with A without even trying to change lane: considering a false speed value, the controller decides that there is no need to change lane, until it becomes too late to prevent the collision. Finally, X-Down1, where the vehicles' sensors use a longitudinal position decreased by 5 meters shows a behavior identical to the base case, while X-Down2, where the estimated positions are decreased by 10 meters, shows that B always overtakes A but a collision may happen. The trace shows that the problem occurs at the junction, between C and the vehicle on the right lane. Indeed, when the vehicles on each side of the junction are actually side by side, they perceive each other 10 m behind and so decide they have enough room to pass, which leads to a collision.

Based on these results, we can say that, in this particular situation, the cases that often cause problems to our controller are those where the speed or position estimated values of the other vehicles are increased in such a way that the vehicle underestimates

how close to collision it is. Of course, doing these checks for a single initial situation does not allow us to determine a general trend about the robustness of our controller. However, applying the same process on various situations, one might be able to evaluate the level of faults a given controller can handle without leading to a collision.

6 Conclusion

We proposed an efficient formal modeling of a complex real time multi-agent system in order to obtain guarantees of its good functioning and robustness. The inherent size of the system made it impossible to address the problem in a direct and naive way, but a suitable discretization and approximation allowed to use model checking exhaustive techniques while limiting the state explosion.

We illustrated the usage of such a modeling to assess the performances of decision making of autonomous communicating vehicles. To do so we implemented controllers and checked their properties, first when evolving in a somehow perfect environment and then in a modified faulty environment. The obtained results show the differences and limits of the considered decision policy, pointing out vulnerabilities. The developed framework is parametric and may easily be adapted to study various kinds of scenarios with a fixed a priori precision. It allows to check a large range of properties, including of course the absence of collisions.

In the future, we plan to extend this approach to cooperative vehicles in order to assess various cooperation strategies in order to improve the safety and increase the fluidity of the traffic, such as negotiations between vehicles. Also, implementing obstacles and human controlled vehicles would be interesting in our approach to quantify decision policy robustness. These extensions would certainly require further abstractions and better exploit system's symmetries.

References

1. R. Alur, C. Courcoubetis, and D. Dill. Model checking in dense real-time. *Information and Computation*, 104(1):2–34, 1993.
2. R. Alur and D. Dill. Automata for modelling real-time systems. In *Proceedings of the International Colloquium on Automata, Languages and Programming (ICALP'90)*, volume 443 of *LNCS*, pages 322–335. Springer-Verlag, 1990.
3. James M. Anderson, Nidhi Kalra, Karlyn Stanley, Paul Sorensen, Constantine Samaras, and Oluwatobi Oluwatola. *Autonomous Vehicle Technology. A Guide for Policymakers.* Research Reports. RAND Corporation, 2016. ISBN: 978-0-8330-8398-2.
4. E. M. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 1999.
5. Google Car official website. https://www.google.com/selfdrivingcar/, 2016.
6. Bertha Benz Drive. http://vra-net.eu/ wiki/index.php?title=Bertha_Benz_Drive, 2015.
7. Mathew O'Kelly, Houssam Abbas, Sicun Gao, Shin'ichi Shiraishi, Shinpei Kato, and Rahul Mangharam. Apex: Autonomous vehicle plan and execution. Technical report, University of Pennsylvania, also in SAE World Congress, April 2016, 2016.
8. Uppaal. http://www.uppaal.org/, 2016.
9. VISLAB Car official website. http://vislab.it/automotive/, 2015.
10. M. Wooldridge. *An introduction to multi-agent systems - Second Edition*. John Wiley & Sons, 2009.

Rule based classifiers for diagnosis of mechanical ventilation in Guillain-Barré Syndrome

José Hernández-Torruco[1], Juana Canul-Reich*[1], and David Lázaro Román[1]

Universidad Juárez Autónoma de Tabasco, División Académica de Informática y Sistemas. Cunduacán, Tabasco, Mexico
(jose.hernandezt, juana.canul)@ujat.mx dlr112h8077@gmail.com
*corresponding author
Note: All the authors equally contributed to this paper.

Abstract. Breathing difficulty is a complication present in almost a third of Guillain-Barré Syndrome (GBS) patients. To alleviate this condition a mechanical respiratory device is needed. Anticipating this need is crucial for patients' recovery. This can be achieved by means of machine learning predictive models. We investigated whether clinical, serological, and nerve conduction features separately can predict the need of mechanical ventilation with high accuracy. In this work, three rule based classifiers are applied to create a diagnostic model for this necessity. JRip, OneR and PART algorithms are analyzed using a real dataset. We performed classification experiments using train-test evaluation scheme. Clinical features were found as the best predictors.

Keywords: Data mining and processing, JRip, OneR, PART, train-test, performance evaluation

1 Introduction

Guillan-Barré Syndrome (GBS) is a condition that damages the peripheral nervous system, i.e. nerves located outside the brain and spinal cord. This causes rapid weakening of muscles accompanied by tingling to pain. In severe cases, it causes fatal paralysis that prevents the person from walking, so that he can remain in bed for the rest of his life. In some severe cases it can cause death [6]. Approximately one-third of patients with this syndrome have breathing problems, and need a mechanism to assist them cite[4]. This need can be predicted with a diagnostic model.

In this study, we investigate the predictive power of clinical, serological, and nerve conduction features separately, from a real dataset. We apply three different rule based classification methods. We chose them for their simplicity in implementation and interpretation. We used as benchmark all the features in the dataset, that is, the group of all the clinical, serological, and nerve conduction features. The performance of the models is evaluated using typical performance metrics like accuracy, sensitivity, among others. Our base metric, however, was

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_22

balanced accuracy since our dataset is imbalanced. We also applied Wilcoxon test to analyze statistical difference among the different models.

Some previous efforts in similar works are: Fourrier, et al. [2] found that, in patients admitted to the ICU (Intensive Care Unit) with GBS and respiratory failure, the combination of a block in motor conduction of the sciatic nerve with the inability to flex the foot at the end of immunotherapy was associated with a prolonged Mechanical Ventilation(MV) with a positive predictive value of 100%, so that, they concluded that it can be a strong argument for performing an early tracheostomy. Sohara, et al. [7], concluded that endotracheal intubation risk factors are increased respiratory rate, decreased VC (Vital Capacity) (20 mL/kg) and VC $<$ 20 mL/kg or 15 mL/kg as well as findings of bulbar dysfunction manifesting as aspiration pneumonia, atelectasis, dysarthria and/or asphyxia. Paul and colleagues [6] identified three clinical variables only to predict the need for MV: the simultaneous onset of motor weakness of the lower and upper limbs as an initial symptom ($p = 0.02$), upper extremity muscle strength $< 3/5$ in the nadir ($p = 0.013$); and bulbar weakness ($p < 0.001$).

This paper is organized as follows. In section 2, we present a description of the dataset, metrics used in the study, a brief description of the classifiers, experimental design. In section 3, we show and discuss the experimental results. Finally in section 4, we summarize results, give conclusions of the study, and also suggest some future directions.

2 Materials and methods

2.1 Data

The dataset used in this work comprises 122 cases of patients seen at Instituto Nacional de Neurología y Neurocirugía located in Mexico City. Data were collected from 1993 through 2002. In this dataset, 41 patients required MV and 81 did not. The original dataset consisted of 156 attributes corresponding to clinical data, results from two Nerve Conduction Studies (NCS), and results from two Cerebrospinal Fluid (CSF) analyses. In this study, we analyze 21 clinical, four CSF and 25 NCS variables. This selection was made according to the findings in the specialized literature. Variables V10 (Days from the onset of breathing difficulty) and V34 (Breathing difficulty) are apriori strong predictors of the MV need. However, these were excluded, since in this study our interest is to investigate whether variables explored by physicians before the onset of the breathing difficulty, if any, allows to accurately predict the necessity of MV. The variable V8 (Date of the onset of the previous event) was excluded and a new variable named *season* was created from it. *Season* takes the values 1 = spring, 2 = summer, 3 = autumn and 4 = winter. The complete lists of each type of variable are shown in Tables 1, 2 and 3.

2.2 Performance measures

We used typical performance measures in machine learning such as accuracy, sensitivity, specificity, AUC (Area under the curve) and Kappa statistic [3,8].

Table 1. List of clinical features

Feature label	Feature name
V4	Age
V5	Sex
V6	Previous event pathology
V7	Days from the onset of muscle strength diminishing or cranial nerve compromise to the previous event
V8 (Season)	Date (Season of the year) of the onset of the previous event
V9	Days from the onset of symptoms to seek medical advice
V21	Weakness
V22	Symmetry
V23	Paresthesia
V24	Upper limb muscle strength
V25	Lower limb muscle strength
V26	Location of symptom onset
V27	Reflexes
V29	Affectation of extraocular muscles
V30	Ptosis
V31	Cerebellar affectation
V32	Ataxic gait
V33	Cranial nerve involved
V37	Complications
V38	Involvement of sphincters
Dysautonomia	Disorder of the autonomic nervous system

We used balanced accuracy as our based metric along all the experiments. This metric is a better estimate of a classifier performance when a unequal distribution of two classes is present in a dataset, as in this case. Balanced accuracy is defined as:

$$BalancedAccuracy = \left(\frac{TP}{TP + FN} + \frac{TN}{FP + TN} \right) /2 \tag{1}$$

where TP = True Positive, TN = True Negative, FP = False Positive and FN = False Negative.

Table 2. List of serological features

Feature label	Feature name
V43	Albuminocytologic dissociation
V44	Cell count
V45	Proteins
V46	Glucose

Table 3. List of NCS features

Feature label	Feature name
V54	Decreased nerve conduction velocity in two or more motor nerve
V55	Nerve conduction block or dispersion
V56	Prolonged distal latency in two or more motor nerve
V57	Absence or extension of F wave in two or more motor nerve
V58	How many Asbury criteria are present?
V59	Which Asbury criteria is not present?
V60	Affectation of motor nerve conduction velocity
V79	Side-to-side difference in left median motor nerve F wave
V98	Side-to-side difference in right median motor nerve F wave
V99	F wave disturbance
V100	Number of disturbed F waves
V101	Motor nerve conduction velocity < 80%
V102	Number of disturbed F waves
V113	Side-to-side difference in left ulnar motor nerve F wave
V124	Side-to-side difference in right ulnar motor nerve F wave
V135	Side-to-side difference in left tibial motor nerve F wave
V146	Side-to-side difference in right tibial motor nerve F wave
V157	Side-to-side difference in left peroneal motor nerve F wave
V168	Side-to-side difference in right peroneal motor nerve F wave
V169	Abnormal sensory conduction velocity in left median nerve
V174	Abnormal sensory conduction velocity in right median nerve
V179	Abnormal sensory conduction velocity in left ulnar nerve
V184	Abnormal sensory conduction velocity in right ulnar nerve
V189	Abnormal sensory conduction velocity in left sural nerve
V194	Abnormal sensory conduction velocity in right sural nerve

2.3 Classifiers

In this study, we apply three different rule based classifiers: JRip, OneR and PART. A brief description of each method is shown below.

OneR OneR stands for One Rule. It generates one rule for each predictor in the dataset, and among all it selects the rule with the smallest total error as the one rule [8].

PART This algorithm obtains rules from pruned partial decision trees using C4.5 [8].

JRip JRip is an implementation of the RIPPER (Repeated Incremental Pruning to Produce Error Reduction) algorithm [1]. JRip builds a set of rules which identifies the classes while minimizing the amount of error. A rule has the form:

if attribute1 <relational operator> value1 <logical operator> attribute2 <relational operator> value2 < ... > **then** decision-value

3 Experiments and Results

3.1 Experimental design

We used the 50-feature dataset, described in section 2.1, for experiments. We added the MV as class variable. We performed experiments using the whole dataset, to use it as a benchmark. Then, we created three different datasets from the original: one containing only clinical variables, another containing only serological variables and the last one containing neuroconduction variables. For each of this datasets, we performed classification experiments using each one of the classifiers mentioned before.

We used train-test evaluation scheme in all cases. We used two-thirds for train and one-third for test. In each classification, we performed 30 runs. For each run, we computed balanced accuracy, accuracy, sensitivity, specificity, Kappa statistic and AUC. Finally, we averaged these quantities across the 30 runs. In each run, we set a different seed to ensure different splits of train and test sets across runs, then we had all classifiers use the same seed at same run. These seeds were generated using Mersenne-Twister pseudo-random number generator [5]. Performing 30 runs with different train and test sets allowed to alleviate the problem of having a limited number of instances in the dataset.

3.2 Results

In this section, we show the results of the experiments. The table below shows the classification using all variables, clinical variables only, serological variables only and neuroconduction variables only. The table shows the average results of each classifier across 30 runs for each classification scenario. Units are in balanced accuracy.

As shown in Table 4, the best results were found using all variables and using only clinical variables, and no major difference was found between these two groups. However, the simplest predictive model, that is, the one using the fewer number of variables is the best among all. This model was the model using only clinical variables with OneR classifier. In fact there was no difference at all in OneR using all or only clinical variables. JRip and PART behaved similarly. Serological and neuroconduction variables by themselves resulted bad predictors, as shown in the the same table.

Table 4. Classification results for each group of variables. Units are in balanced accuracy averaged across 30 runs.

Classifier	All	Clinical	Serological	Neuroconduction
JRip	0.8908	0.8884	0.5462	0.5225
OneR	**0.9119**	**0.9119**	0.5049	0.5456
PART	0.8813	0.8833	0.5260	0.5775

In Table 5, we show the best rules across 30 runs for each classifier. We can observe that in the test set, each of these rules obtained a balanced accuracy above 0.95. Another highlight is the presence of the variable Complications (v37) in all rules.

Table 5. Best rules for each classifier. Derived from clinical features.

Classifier	Best rule	Balanced accuracy in test set
OneR	(Complications == Low respiratory tract infection) => VM = 1 (Complications == Eschar) => VM = 2 (Complications == Depression) => VM = 2 (Complications == Other) => VM = 1 (Complications == No) => VM = 2 (Complications == Anxiety) => VM = 2 (Complications == Low respiratory tract infection and anxiety) => VM = 1 (Complications == Low respiratory tract infection and eschar) => VM = 1 (82.0/6.0)	1
JRip	(Complications == Low respiratory tract infection) => VM=1 (20.0/0.0) (Dysautonomia == YES) AND (Upper limb muscle strength == 6) => VM=1 (3.0/0.0) => VM=2 (59.0/5.0)	1
PART	(Complications == Anxiety OR Complications == Low respiratory tract infection and anxiety OR Complications == Low respiratory tract infection and eschar) AND (Dysautonomia == YES) => VM=2 (51.0/2.0) (Involvement of sphincters == YES) AND (Complications == Low respiratory tract infection) => VM=1 (20.0/0.0) (Cerebellar affectation == YES) AND (Season == SPRING OR Season == SUMMER) => VM=1 (4.0/0.0) (Lower limb muscle strength <= 5) => VM=2 (4.0/0.0) => VM=1 (3.0/1.0)	0.9815

We investigated if there was any statistical significant difference of the best classifier, OneR, with respect to JRip and PART. For this purpose, we applied Wilcoxon signed rank test. We used all the balanced accuracy found in each of the 30 runs, using clinical variables only. We used a significance level of 0.05. As

shown in Table 6, there was a statistically significant difference in the balanced accuracy between OneR and JRip across 30 runs, and also between OneR and PART. This means that, the model created with OneR, using clinical variables only, is in fact the best model of them all.

Table 6. Wilcoxon signed rank test results.

Hypothesis	p value	H0 result
H0: There is no statistically significant difference in the balanced accuracy between OneR and JRip across 30 runs H1: There is a statistically significant difference in the balanced accuracy between OneR and JRip across 30 runs	0.00069	rejected
H0: There is no statistically significant difference in the balanced accuracy between OneR and PART across 30 runs H1: There is a statistically significant difference in the balanced accuracy between OneR and PART across 30 runs	0.00139	rejected

In Table 7, we show the classification results of clinical features for each classifier. The standard deviation (sd) is also shown. OneR obtained the best classification result. JRip and PART obtained similar results. As we can see, all classifiers obtained a balanced accuracy above 0.88. Another relevant fact is that all models resulted to be more specific than sensitive, this could be a consequence of the imbalanced characteristic of the dataset, this requires more investigation.

Table 7. Classification results using only the clinical features. The results are shown in balanced accuracy over 30 runs. The highest value appears in bold. Standard deviation (sd) is also shown.

Classifier	Balanced accuracy	Accuracy	Sensitivity	Specificity	Kappa	AUC
OneR	**0.9119**	0.9042	0.8641	0.9235	0.8493	0.9119
sd	0.0425	0.0496	0.0919	0.0673	0.0817	0.0425
JRip	0.8884	0.9167	0.8077	0.9691	0.8034	0.8884
sd	0.0470	0.0379	0.1025	0.0542	0.0877	0.0470
PART	0.8833	0.9017	0.8308	0.9358	0.7744	0.8833
sd	0.0531	0.0473	0.0956	0.0583	0.1063	0.0531

4 Conclusions

In this study, we aimed at building and comparing predictive models for the MV necessity in GBS patients. We applied rule based classification methods, specifically JRip, OneR and PART. We investigated the predictive power of clinical, neuroconduction and serogological features separately. For experiments we used a real dataset. After the analysis we found that clinical features are the best predictors and OneR the best classifier for this task. We found also feature Complications (v37) is a very important variable to predict MV necessity in GBS patients.

From the medical point of view, this finding could help physicians to promptly and accurately identify GBS patients with mechanical necessity. It is desirable for them to have clinical variables only in the prediction model. So, it is a good finding to show that our clinical variables reached the highest accuracy. However, clinical experimentation by physicians using the results found in this study are part of our ongoing research.

References

1. W.W. Cohen. Fast effective rule induction. In *In Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.
2. Fourrier F., Robriquet L., Hurtevent J.F., and Spagnolo S. A simple functional marker to predict the need for prolonged mechanical ventilation in patients with guillain- barr syndrome. *Critical Care*, 15: R65, 2011.
3. J. Han, M. Kamber, and J. Pei. *Data mining: concepts and techniques*. Morgan Kaufmann, San Francisco, USA, 2012.
4. Satoshi Kuwabara. Guillain-barré syndrome. *Drugs*, 64(6):597–610, 2004.
5. M. Matsumoto and T. Nishimura. Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Trans. Model. Comput. Simul.*, 8(1):3–30, January 1998.
6. B.S. Paul, R. Bhatia, K. Prasad, M.V. Padma, M. Tripathi, and M.B. Singh. Clinical predictors of mechanical ventilation in guillain-barre syndrome. *Neurol India*, 60(2):150–3, Mar-Apr 2012.
7. E. Sohara et al. Guillain-barr syndrome in two patients with respiratory failure and a review of the japanese literature. *Journal of Thoracic Disease*, 4(6):601–607, 2012.
8. I.H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and techniques*. Morgan Kaufmann, third edition, 2011.

CKMultipeer: Connecting Devices Without Caring about the Network

Jose-Enrique Simó-Ten, Eduardo Munera, Jose-Luis Poza-Lujan, Juan-Luis Posadas-Yagüe, and Francisco Blanes

University Institute of Control Systems and Industrial Computing (ai2) Universitat Politècnica de València (UPV). Camino de vera, s/n. 46022 Valencia (Spain)
jsimo@ai2.upv.es, emunera@ai2.upv.es, jopolu@ai2.upv.es, jposadas@ai2.upv.es, pblanes@ai2.upv.es

Abstract. In the current context of distributed systems, the communications have moved from a model based on connected nodes, to a model that must connect processes, or including connect data. There are many paradigms, mechanisms and architectures to give support to these distributed systems. Most of these mechanisms are extensions or evolutions of distributed node based architecture where the knowledge about the physical node is necessary. To isolate communications from the physical aspects of distributed systems a middleware, centered on the data, and oriented to connect processes has been developed. The middleware, called "CKMultipeer", allows processes to communicate data between them. Communication is integrated without worrying about the network configuration and management. This article describes the implementation of the middleware, as well as the tests developed. Through the design of a graphical interface, the communications performance is analyzed.

Keywords: distributed system, middleware, peer to peer

1 Introduction

Among the different paradigms to manage distributed systems, those that offer a high level of connectivity between nodes have increased their presence. The high level of connectivity is referred, mainly, to the large amount of data needed, currently, in the distributed systems. Data is interchanged between processes running in nodes. The first distributed systems were created to connect nodes locating them by means a network direction. These distributed systems cover a broad range of applications. From the personal and domestic connection needed from the Internet of Things (IoT)[1] until the critical communications of the Smart Factories (also known as Industry 4.0) [2]. One of the main needs of these systems is the need of providing connected solutions without caring of the technical aspects for network configuration and management. The proposed work, developed at the Institute of Control Systems and Industrial Computing (ai2), provides a programming model for data-centered control applications called Control Kernel Multipeer (CKMultipeer). The CKMultipeer aims to automatically

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_23

detect local and distributed networked processes, establish a permanent connection between them, and perform a reliable and efficient data exchange. In order to provide these capabilities, the next objectives are established:

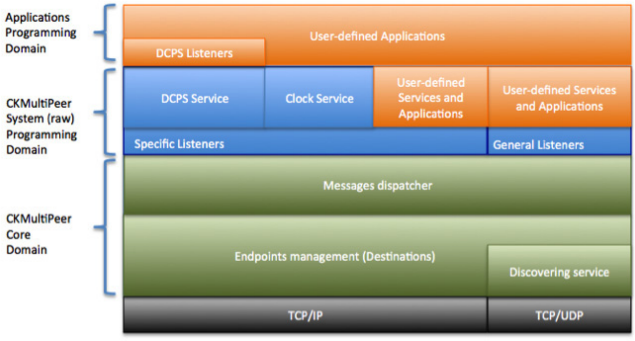


Fig. 1: Multipeer.

- Develop an asynchronous communication, which relies on a TCP/IP implementation, without payload size restrictions.
- Integrate a process discovery service implemented on a TCP/UDP protocol that doesn't require connection. The proposed discovery protocol must automatically detect the peers that belong to the communication group and establish a connection between them.
- Establish a message priority mark, and ensure that exchanged messages offer microsecond level timestamp.
- Develop a monitor interface in order to check and validate the suitability of this proposal.

The communication between peers must be performed without the requirement of a specific infrastructure such as a centralized middleware or data servers. CKMultipeer solution is integrated as a part of the process through a library or a package. These characteristics allows to use CKMultipeer in JAVA or C++ over POSIX infrastructure. These characteristics configure a open, and flexible system that can be adapted to the specifics needs of a project.

This paper is organized as follows. Section 2, exposes a review of similar systems. Section 3 describes the details of the CKMultipeer design. Section 4 describes the software developed in order to monitor and validate the CKMultipeer. Finally, Section 5 summarizes the advantages of the proposed system.

2 Related work

Distributed Systems usually implements communication middleware, which extends the execution framework in order to provide device connection and data ex-

change mechanisms. It can be found many communication solutions such as Java Message Service (JMS) [3], Internet Communication Engine (ICE) [4], Common Object Request Broker Architecture (CORBA) [5], or Distributed Data System (DDS) [6]. Some of them, like JMS, provide a message-centric architecture, while some others, such as DDS, provide a data-centric approach. In spite of this, all of them aim to provide reliable and flexible mechanisms for asynchronous data exchange. JMS enables distributed communication by adding a common API for the development of message-based applications in Java. DDS provide a publish-subscriber platform independent solution. ICE and CORBA frameworks, both of them, make use of an Object Request Broker (ORB) that defines the interface for exchanged objects. Common ORB solutions are based on "The ACE ORB" (TAO) [7] which relies on the Adaptive Communication Environment (ACE) framework [8]. In order to bring the system a fault-tolerance and/or adaptation mechanisms it is need to offer the proper mechanisms to measure the performance. For that reason, some communication frameworks, as the introduced TAO, or DDS between many other, include Quality of Service (QoS) mechanisms [9]. As introduced in [10], the design of real-time QoS-aware mechanisms allows to accurately evaluate the service performance. The QoS mechanisms create a special framework to manage the communications also known as QoS policy [11].

3 CKMultipeer Description

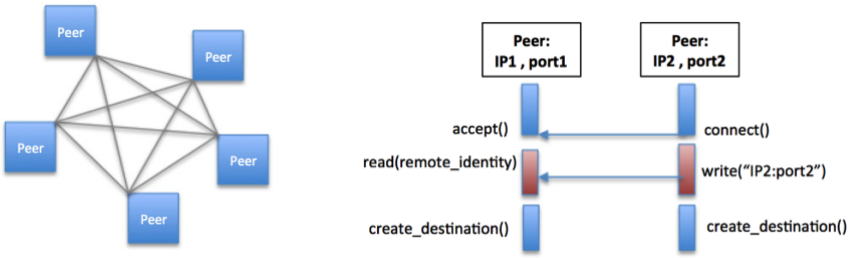
3.1 Peer Connexion

Each member of the group establishes TCP/IP connections with the rest of the members by forming a connection "mandala" as shown in Fig. 2(a). In order to setup this network, each member within the network accepts connections on a certain port through a server thread. Whenever this port is busy and the connection cannot be established, it will retry to connect to the next port number until the connection is finally established or a maximum number of tries is reached. Once the connection is established, this pair establishes its identity as a 64-bit Globally Unique Identifier (GUID) computed by combining the IP address (32 bits unsigned) and the port number (16 bits).

The GUID of a pair, not only identifies it among the network in a single way, but also is used to establish an interaction order between the pairs. Therefore, an ordered connection structure is set. Given a pair with a certain GUID, it will establish the connection with those pairs with a grater GUID, and let the pairs with a lower GUID to request the connection. The interaction between pairs during the connection procedure is detailed in Fig. 2(b).

3.2 Peer Discovery

Whenever a new CKMultipeer object is started, it establishes a multicast broadcast channel by choosing an IP address (type "D" from 224.0.0.0 to 239.255.255.255)



(a) Peer connection mandala (b) Connection protocol

Fig. 2: CKM network topology and connection mechanism.

and a port. Therefore, the "group identity" is set as the combination of the IP address and the broadcast port number. Each member of the group broadcasts their identity by sending their GUID through the broadcast channel. In the same way, the identity of all the other group members is received through this channel. Every time a pair receives the identity of a new one, it broadcasts its own identity in order to ensure that the new pair of the group is aware of it. Once there are no new pairs in the group, the broadcast channel remains silent. However, every 5 seconds its identity is sent again as a "heartbeat" for verifying that the current network setup is still valid. A pair that receives the identity of a new pair within the group must execute a connection sequence (as previously described in Fig. 2) in order to setup the CKMultipeer communication infrastructure.

3.3 Messasge Types

In order to transmit information through CKMultipeer network, it is necessary to define a message structure. Therefore, a set of different types of messages has been developed. Each message type is implemented as a class that extends the abstract base class "Message". As a result, all the message types share a common format consisting of a prologue, a header, and a load.

Current message support implementation includes next types: PLAIN, TOPICDATA, TOPICLINKS, SYNCTIME, and PROXYWRAPPER. A detailed description of these types can be found in Fig. 3. Therefore, every CKMultipeer object can implement services that filter messages by type. As an example, the SYNCTIME message type is managed by the clock synchronization service, while TOPICDATA and TOPICLINKS messages are related with the publication/subscription service. A regular CKMultipeer user that just wants to transmit or receive information must use the PLAIN message type.

3.4 Listeners

A MultiPeerListener class is implemented in order to receive CKMultipeer messages. The listener interface implements callback mechanisms for peer connec-

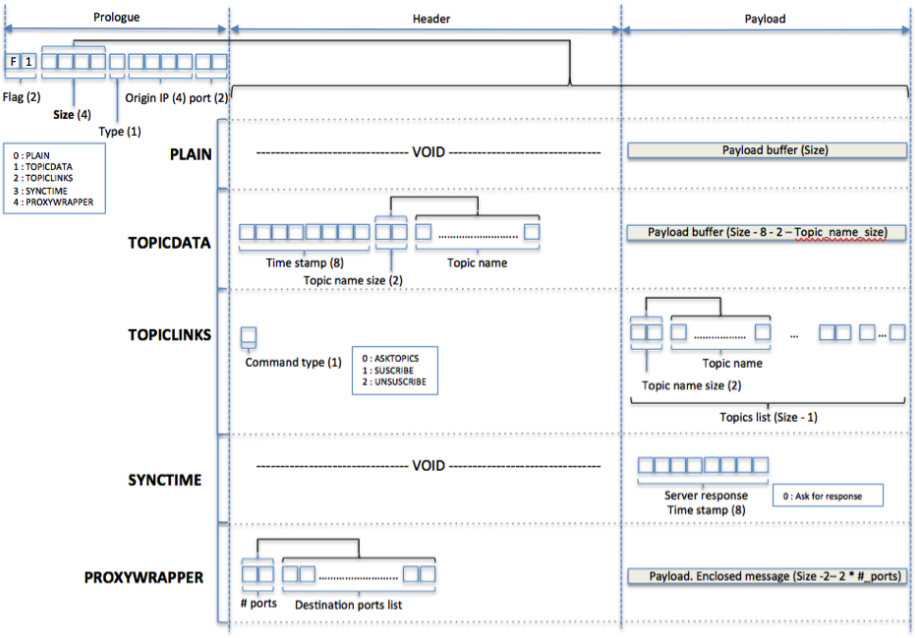


Fig. 3: Message Types.

tion/disconnection and message management. Current message callback implementation pays attention to the life cycle of the memory. After invoking the callback it automatically destroys the message for freeing the memory space. Whenever is required to store the message information, it must be managed within the callback.

3.5 Using CKMultipeer

In order to make use of the CKMultipeer capabilities first step is to create a CKMultipeer object. During this step, the MultiPeerConf configuration object can be detailed. This object contains the following information: InterfaceName, interfaceAddress, initialPort, multicastAddress, and MulticastPort. The configuration can be modified through the "getters" and "setters" methods. Despite of this, the interfaceAddrees parameter can not be modified since this address is determined by the network interface.

The CKMultipeer communications can only work on network interfaces that support Multicast traffic. Therefore, network devices (such as adapters and routers) must be properly configured so the multicast traffic can reach all points among the network in order to use CKMultipeer. Whenever a device is not connected to any network, CKMultipeer can communicate local processes by using the loopback network interface (127.0.0.1).

The CKMultipeer communications layer is characterized as a publisher/subscriber topology. Therefore the "Topic" and "TopicScope" classes manage the TOPICDATA and TOPICLINKS message types. These messages are managed through the publisher services and the subscribers, which must implement the "TopicListener" interface.

4 Experiments and Results

In order to monitor and test the performance of the CKMultiPeer it has been implemented a visualizer as showed in Fig. 4. Even though, the visualizer has been implemented in Java, it is compatible with both C++ and Java CKMultipeer implementations. This visualizer allows listing all the connection pairs within the network and the related topic information. This interface also allows monitoring the topic information by analyzing its data and temporal parameters, which includes: payload type, priority, payload size, apparent period, and delay. Furthermore, it offers specific data viewers in order to check the message content. Most common view provides a topic console in which the topic data is serialized, but also offers specific views for images, joystick commands, IMU measures, and a 2D grapher.

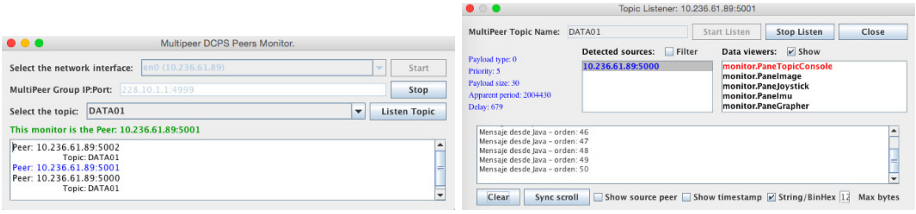


Fig. 4: CKMultiPeer Visor Interface.

The communication performance values have been analyzed along a set of experiments. First of all, it must be detailed that this system implements a NTP clock synchronization that provides a clock skew lower than 1ms. As a result, the average communication latency is characterized as 30us for inter-process communication, 300us for communication between local processes, and 1.5ms between distributed processes among the network devices. These performance values are depicted in Fig. 5. Moreover, Fig. 5 also provides a comparison between the CKMultipeer performance and some of the communication middlewares introduced in Section 2. As can be observed, CKMultipeer communication provides a similar performance for the local communication obtained when dealing with JMS or CORBA, but is improved for the distributed one. Even though the DDS performance is always better, the CKMultipeer provides additional advantages such as being a multi-architecture solution (C++ and Java) or providing an automatic peer discovery and connection mechanisms.

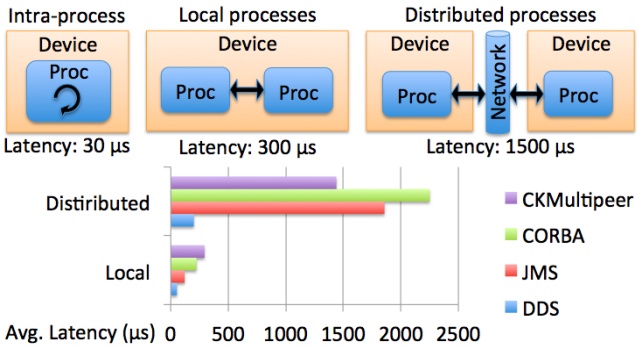


Fig. 5: Performance comparison between CKMultiPeer and other communication middlewares.

5 Conclusions and Future Work

This paper presents a novel publisher/subscriber middleware for real time communications known as CKMultipeer. Among its main characteristics, CKMultipeer implementation provides a discovery protocol mechanism that allows new nodes to be automatically incorporated into the network group without depending on any initial configuration or central node mediation. Since no central node is required every node within the system is characterized as a fully autonomous device. Whenever the network fails or a process is reset, the CKMultipeer implementation guaranties an automatic reconnection. Besides, CKMultipeer incorporates a Network Time Protocol (NTP) to synchronize all the node clocks of the group. So, messages time stamps is used in to be sort and match the information along time.

Its access interface is based on the DCPS model and it has been implemented in C++ language for the Linux platform and in Java language for multiplatforms. As a result, every application can access to the logic data bus by implementing the CKMultipeer API. This API allows application processes to write and read the topic information among the network. Uncoupling senders and receivers allows the CKMultipeer to guarantee a popper topic publication rate. Therefore, topic readings are performed through an asynchronous callback event.

The system has been validated and tested to provide the expected performance. Because of the characteristics of CKMultipeer, it is characterized as a suitable solution for distributed system developments that involve many different platforms. Thanks to its mechanisms the CKMultipeer fills the needs for distributed application communication in a easy, transparently and efficient way. CKM also guarantees real-time transmission rates and provides messages timestamp. Because of this, CKMultipeer communications are suitable for im-

plementing Enterprise Service Bus (ESB) based solutions without needing any additional middleware or specific configuration.

Acknowledgements

Work supported by the Spanish Science and Innovation Ministry MICINN: CICYT project M2C2: "Codiseño de sistemas de control con criticidad mixta basado en misiones" TIN2014-56158-C4-4-P and PAID (Polytechnic University of Valencia): UPV-PAID-FPI-2013.

References

1. Soma Bandyopadhyay, Munmun Sengupta, Souvik Maiti, and Subhajit Dutta. A survey of middleware for internet of things. In *Recent Trends in Wireless and Mobile Networks*, pages 288–296. Springer, 2011.
2. Eduardo Munera, Jose-Luis Poza-Lujan, Juan-Luis Posadas-Yagüe, Jose Simo, J Francisco Blanes, and Pedro Albertos. Control kernel in smart factory environments: Smart resources integration. In *Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2015 IEEE International Conference on*, pages 2002–2005. IEEE, 2015.
3. Mark Hapner, Rich Burridge, Rahul Sharma, Joseph Fialli, and Kate Stout. Java message service. *Sun Microsystems Inc., Santa Clara, CA*, page 9, 2002.
4. M Henning and M Spruiell. Distributed programming with ice. Technical report, ZeroC Inc., 2003.
5. Object Management Group (OMG). The Common Object Request Broker (CORBA): Architecture and Specification. Technical report, Object Management Group, 1995.
6. Object Management Group (OMG). *Data Distribution Service for Real-time Systems.* Version 1 edition, 2007.
7. DC Schmidt. TAO, The Ace Orb. Technical report, DOC group, Washington University, 2007.
8. DC Schmidt. The adaptive communication environment. Technical report, DOC group, Washington University, 1994.
9. Cristina Aurrecoechea, Andrew T Campbell, and Linda Hauw. A survey of qos architectures. *Multimedia systems*, 6(3):138–151, 1998.
10. Woochul Kang, Sang Hyuk Son, and John A Stankovic. Design, implementation, and evaluation of a qos-aware real-time embedded database. *Computers, IEEE Transactions on*, 61(1):45–59, 2012.
11. José L Poza, Juan L Posadas, and José E Simó. From the queue to the quality of service policy: A middleware implementation. In *International Work-Conference on Artificial Neural Networks*, pages 432–437. Springer, 2009.

Disease diagnosis on short-cycle and perennial crops: An approach guided by ontologies

Katty Lagos-Ortiz[1], José Medina-Moreira[1], José Omar Salavarria-Melo[1], Mario Andrés Paredes-Valverde[2], Rafael Valencia-García[2]

[1]Universidad Agraria del Ecuador, Avenida 25 de julio, Guayaquil, Ecuador
[2]Facultad de Informática, Universidad de Murcia, Campus Espinardo, 30100, Murcia, Spain

klagos@uagraria.edu.ec, jmedina@uagraria.edu.ec, jsalavarria@uagraria.edu.ec, marioandres.paredes@um.es, valencia@um.es

Abstract. It is extremely important that farmers regularly monitor their crop looking for symptoms that may reveal the presence of diseases. However, sometimes farmers have no access to information that helps them to respond to questions such as: what is wrong with their crop? and what can they do to deal with the problem? This situation could cause them to lose their crops, which in turn represents economic losses. Nowadays, there are solutions focused on the automatic diagnosis of diseases, including human diseases and diseases of specific crops such as maize. However, there is a lack of solutions focused on the diagnosis of diseases of short-cycle and perennial crops. In this sense, we propose an ontology-based solution for helping farmers to diagnose disease of such kind of crops from a set of symptoms perceived by farmers. For this purpose, our solution implements a rule-based engine that can diagnose a disease from the symptoms provided. The ontology and rule-based engine were designed in conjunction with a group of experts in plant pathology. Our proposal was evaluated in conjunction with farmers from the Costa Region of Ecuador achieving encouraging results.

Keywords: ontology, plant disease diagnosis, crop

1 Introduction

On a global scale, plant diseases cause an estimated $38 billion in annual losses [1]. A plant disease is any abnormal condition that alters the appearance or functions of a plant and may reduce the quality and/or quantity of the harvested product [2]. Considering the above facts, it is extremely important that farmers regularly and systematically monitoring their crop looking for signs or symptoms that may reveal the presence of a disease. However, sometimes it may be late to counteract the disease because farmers have no access to experts' knowledge-based technologies that help them to respond to questions such as: *what is wrong with his/her crop?* and *what can*

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_24

he/she do to deal with the problem? A proper diagnosis is very important to prevent the problem on other crops as well as to prevent the problem in the future.

Plant pathology is the scientific study of the environmental conditions and the living entities that cause diseases in plants as well as the methods of preventing or controlling disease and alleviating the damage it causes [3]. One of the main goals of this science is to increase the knowledge about causes and the development of plant diseases. Therefore, there is a need to make this kind of information available to farmers in a clear and intuitive way aiming to help them to make a proper diagnosis and then counteract the disease with proven treatments provided by experts. The Semantic Web is an extension of current one, in which information is given a well-defined meaning, better-enabling computer and people to work in cooperation [4]. Semantic Web allows people to create data stores on the Web, build vocabularies, and write rules for handling this data. One of the pillars of the Semantic Web are the ontologies. An ontology can be defined as a formal and explicit specification of a shared conceptualization [5]. The ontologies have been applied in the modelling of knowledge in domains such as recommender systems [6], natural language interfaces [7], finances [8] and cloud services [9], among others.

In this work, we propose an ontology-based solution for helping farmers to diagnose diseases of short-cycle and perennial crops from a set of symptoms. This solution is guided by both: an ontology that describes the phytopathology domain, more specifically, the concepts related to diseases of short-cycle and perennial crops, such as symptoms and the environmental conditions and living agents that produce them, and a rule-based engine that allows to diagnose a plant disease from a set of symptoms provided by the farmers. The remainder of this work is organized as follows. Section 2 describes the most relevant works regarding human and plant disease diagnosis. The formalization of our research work is presented in section 3. Section 4 outlines a set of experiments performed to evaluate the effectiveness of our proposal. Finally, conclusions and future work are discussed in Section 5.

2 Related works

Recently, many researchers have focused on the automatic diagnosis of diseases. For instance, in the context of human diseases, SeDeLo [10] represents a clinical decision support system based on Semantic Web technologies that is able to infer human diseases from symptoms, signs and laboratory tests represented as logical descriptions. SeDeLo is based on single description logics [11]. Meanwhile, in [12] and [13] the authors present two different systems that allow diagnosing the Alzheimer disease. To that end, these systems use ontologies and a set of rules produced by experts. On the other hand, in [14] the authors present a recommender system that combines the use of an ontology for describing anti-diabetic drugs, patient tests, etc., and SWRL [15] rules to suggest the most appropriate diabetes medication. DO4MG (Domain Ontology for Mass Gatherings) [16] is an ontology that models individual experiences and knowledge of managerial personnel of medical emergency services. This ontology serves as the basis for the implementation of a case-based reasoning system for

emergency medical management in mass gatherings. Finally, in [17] the authors present a system that enables to diagnose diseases that affect maize plants. This system is based in two main elements, a maize diseases ontology that describes concepts such as diseases, symptoms and parts of the maize plant; and SWRL and SQWRL [18] that allow defining the rules of maize disease diagnosis. Furthermore, it uses Jess (Java Expert System Shell) [19], an inference engine based on Java language, as the reasoning engine in the SWRL reasoning rules. Most works described in this section use ontologies and a set of rules to diagnose human diseases such as Alzheimer or to suggest diabetic medication. The rules used by these proposals are produced by experts and based on the semantic knowledge concerning signs, laboratory tests, drugs, patient tests, or individual experiences. Furthermore, most of these works use description logic and SWRL to perform their tasks. Based on these facts, it is difficult to establish which work is better than other works due to each of them operates in a different domain that varies in their size and complexity. However, none of them are focused on the diagnosis of plant diseases, specifically, diseases of short-cycle and perennial crops. Hence, in this work, we propose a system for plant disease diagnosis that combines the use of ontologies for describing the plant pathology domain, and a rule-based engine that diagnoses a plant disease from a set of symptoms. Furthermore, it provides treatments and recommendations suggested by experts, which are formalized in the plant pathology ontology.

3 Plant disease diagnosis system guided by ontologies

In this paper, we propose a system able to diagnosing a plant disease based on a set of symptoms. Fig.1 depicts the overall architecture of our system which is decomposed into two constituent parts: (1) an ontology that describes the plant pathology domain through concepts related to the causes (living entities and environmental conditions), symptoms, treatments and recommendations for plant diseases, and (2) a rule-based engine that can diagnoses a disease from a set of symptoms. Based on such diagnosis, the system provides specific treatments or recommendations that help farmers to counteract the disease. Following, we provide a wider description of these components.

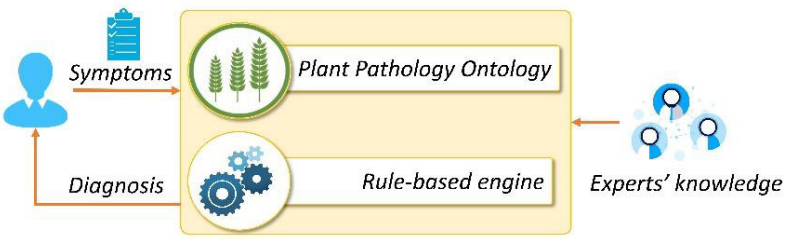


Fig. 1. A plant diseases diagnosis system guided by ontologies.

3.1 Plant pathology ontology

The Plant Pathology Ontology (hereafter referred as PPO) was designed in conjunction with a group of researchers from the Agrarian University of Ecuador. The ontology design process involved a literature review about ontologies focused on aspects such as plant anatomy, symptoms and causes of plant diseases. As result of this task, we identified two outstanding ontologies: PPIO (Plant-Pathogen Interactions Ontology) [20] which served as basis for the specification of causes of plant diseases, and the Ontology for Plan Protection [21], which helped us with the definition of concepts such as plant parts, disorders and environmental conditions. Some classes contained in PPO are:

- Disease. This class contains a taxonomy of diseases of short-cycle and perennial crops. The first group includes diseases such as downy mildew, gray ear rot, pyricularia, cucumber mosaic, etc. Some diseases of perennial crops are black sigatoka, phytophthora, botrytis, anthracnose, among others.
- Cause. This class contains all living entities and environmental conditions that cause diseases in plants. Some of the living entities described are bacteria, fungi, nematodes, parasitic plant. Examples of environmental conditions are light intensity, humidity, rainfall, soil temperature, among others.
- Plant. This class contains a taxonomy of short-cycle and perennial crops. Some short-cycle crops are beans, corn, peanut, rice pepper. Some perennial crops are banana, pineapple, grape, sugarcane, mango, to mention but a few.
- Symptom. The symptom class contains all the symptoms available in the system. A symptom is a phenome accompanying something, in this case, a plant disease, and is regarded as evidence of its existence [22]. Some of the symptoms described by this ontology are: shoot blight, leaf blight, fruit spot, fruit rot, leaf spot, root rot, canker, gray mold, small chlorotic spots, glossy black stain, among others.

The PPO ontology also contains the classes *Treatment*, *Recommendation* and *Pesticides*. The first two classes are related to a specific disease, meanwhile, the third class is related to the *Treatment* class. These classes help the system to provide advices about how to counteract the diagnosed disease. Regarding ontology properties, the PPO ontology contains some object properties to establish the relations between the instances of the classes above described. Some of these properties are:

- hasSymptom. It connects Disease class with Symptom class. For instance, thanks to this property we could specify that the “Black Sigatoka” disease has the symptom “chlorotic spots”.
- causedBy. Property to connect a disease with a cause, whether environmental condition or living entity. For example, through this property the system can determine that the “Black Sigatoka” disease is caused by the fungus “Mycosphaerella fijiensis”.
- hasTreatment. It connects Disease class with Treatment class. By means of this property, it is possible to provide users disease management advice, for instance,

the system can recommend the use of the fungicide "Mancozeb" to control the "Black Sigatoka" disease.

The PPO ontology also contains the properties *isSymptomOf* and *isTreatmentOf*, which are the inverse properties of the ones above. In this way, we could say that the symptom "chlorotic spots" *isSymptomOf* the "Black Sigatoka" disease. Finally, it must be said that the PPO ontology also contains data type properties that allow establishing values such as code, name, etc.

3.2 Rule-based engine

Ontologies are based on first-order predicate logics as underlying knowledge representation and reasoning paradigm. Hence, in this work we take advantage of this feature to define a set of rules for inferring plant diseases from the knowledge described in the PPO ontology. Rule-based inference engines are used in domains such as supporting clinical decision support system, access control mechanism [23]. In this sense, our proposal implements a rule-based engine oriented to the plant disease diagnosis. For this purpose, it implements Pellet reasoner [24], which provides support for reasoning with individuals and user defined datatypes. Furthermore, Pellet allows inferring logical consequences from a set of asserted facts or axioms [25]. In this work, a set of SWRL [15] rules were implemented aiming to define plant diseases according to its symptoms. For instance, the definition of Black Sigatoka of bananas is represented as follows:

```
disease(?x) ∧ hasSymptom(?x, "tiny, chlorotic spots") ∧
hasSymptom(?x, "leaf blackened") ∧ hasSymptom(?x, "black,
globose fruiting bodies") ∧ hasSymptom(?x, "black streak-
ing in the leaves") →isDisease(?x, BlackSigatoka)
```

The above rule specifies that Black Sigatoka will be inferred when property *hasSymptom* matches with the symptoms that define the disease, in this case, *"*tiny, chlorotic spots*"*, "leaf blackened", "black, globose fruiting bodies" and "black streaking in the leaves". It should be mentioned that, due to symptoms are not usually binary related to a disease, our proposal defines different rules that can be applied for a set of given symptoms. Finally, once disease has been diagnosed, the system provides users the treatments or recommendations that help them to counteract such disease. To that end, it uses the *hasTreatment* and *hasRecommendation* properties.

4 Evaluation and results

This section describes a set of experiments that were performed to validate the effectiveness of our approach to generate a correct diagnosis from a set of symptoms. This process involved the participation of ten farmers from the Costa Region of Ecuador with which the Agricultural University of Ecuador have been collaborating in the

development of technological solutions for the diagnosing and prevention of plant diseases. The evaluation process was decomposed into four steps:

1. Farmers taking part in this task were asked to provide a set of symptoms they perceive when a specific disease is presented. For the purposes of this study, the short-cycle crops considered were beans, peanut, maize, rice and cucumber. Some of the diseases of short-cycle crops described by means of rules are anthracnose, downy mildew, bean golden yellow mosaic; early leaf spot, bacterial wilt, holcus spot, and charcoal rot among others. Regarding perennial crops, we considered banana, sugarcane, pineapple, grape and passion fruit. Some of the diseases described through rules are black sigatoka, bract mosaic, Panama disease, gumming disease, mottled stripe, to mention but a few. As result of this task, we obtained 100 sets of symptoms, specifically 10 sets for the diseases of each plant considered.
2. Each set of symptoms was provided to the system aiming to obtain a diagnosis as well as the corresponding treatments and recommendations.
3. The results provided by our proposal were validated by a group of experts outside the research team. This group has a great experience in plant disease management, specifically, diseases of short-cycle and perennial crops.
4. To measure the effectiveness of our proposal we used the metrics precision (P) and recall (R) [26], and F-measure (F) [27], where true positives are the sets of symptoms correctly associated to a disease *X*, false positives are those sets that were incorrectly related to the disease *X*, and false negatives are those sets that were related to a disease other than *X*. The evaluation results grouped by crop are shown in Table 1.

Table 1. Evaluation results

Short-cycle	P	R	F		Perennial	P	R	F
Beans	0.7727	0.8500	0.8095		Banana	0.6087	0.7000	0.6512
Peanut	0.6818	0.7500	0.7143		Sugarcane	0.8421	0.8000	0.8205
Maize	0.8235	0.7000	0.7568		Pineapple	0.8235	0.7000	0.7568
Rice	0.8750	0.7000	0.7778		Grape	0.7895	0.7500	0.7692
Cucumber	0.6957	0.8000	0.7442		Passion fruit	0.5909	0.6500	0.6190
Avg.	0.7697	0.7600	0.7605			0.7309	0.7200	0.7233

As can be observed in Table 1, there are no significant differences among the results obtained by the diseases of each crop considered. This fact can be interpreted as a good effectiveness of our proposal for diagnosing plant diseases of several short-cycle and perennial crops. Furthermore, in this table, we can observe that the highest values were obtained for the diseases that affect sugarcane plants. Meanwhile, the lowest values were obtained for the diseases that affect passion fruit plants. We ascribe these variations to the fact that there are diseases that share multiple symptoms which makes difficult to do a proper diagnosis. Aiming to address this issue, it is necessary to include more symptoms that help to differentiate this kind of diseases. Furthermore, we think that the set of rules implemented in this version could be improved aiming to be more precise regarding the part of the plant where the symptom is per-

ceived. The hypothesis is that this improvement could help us to deal with the above issue.

5 Conclusions and future work

This work presented an ontology-based solution for automated plant diseases diagnosis. Our proposal benefits from ontologies in two main ways. First, they allowed to formalize the plant disease domain, more specifically, the causes, symptoms, treatments and recommendations for diseases of short-cycle and perennial crops. Second, it represents a source of computable knowledge that is exploited by the rule-based engine for disease diagnosis. Our approach obtained encouraging results. However, we are aware that our solution must be improved including more specific symptoms that help us to differentiate the diseases in a better way. Also, as was mentioned in the previous section, we think that the rules implemented so far can be more specific regarding the part of the plan where the symptom is observed, thus allowing to differentiate diseases that share multiple symptoms. As future work, we believe that a medical system methodology evaluation such as the presented in [28] could be used for assessing the efficiency of our proposal through feedback of farmers. Also, we are considering the use of image processing technologies to automatically identify observable patterns on different parts of the plant that could correspond to some of the symptoms described by the ontology.

Acknowledgments

Mario Andrés Paredes-Valverde is supported by the National Council of Science and Technology (CONACYT), the Public Education Secretary (SEP) and the Mexican government. Finally, this work has been also partially supported by the Spanish Ministry of Economy and Competitiveness and the European Commission (FEDER / ERDF) through project KBS4FIA (TIN2016-76323-R).

References

1. K. Leonberger, K. Jackson, R. Smith, and N. W. Gauthier, “Plant Diseases [2016],” *Agric. Nat. Resour. Publ.*, Mar. 2016.
2. A. Deshpande, “A Review on Preharvesting Soybean Crop Pests and Detecting Techniques,” *Int. J. Adv. Res. Comput. Sci. Manag. Stud.*, vol. 2, no. 2, 2014.
3. G. N. Agrios, *Plant Pathology*. Elsevier, 2012.
4. T. Berners-Lee, J. Hendler, O. Lassila, and others, “The semantic web,” *Sci. Am.*, vol. 284, no. 5, pp. 28–37, 2001.
5. R. Studer, V. R. Benjamins, and D. Fensel, “Knowledge engineering: Principles and methods,” *Data Knowl. Eng.*, vol. 25, no. 1, pp. 161–197, Mar. 1998.
6. L. O. Colombo-Mendoza, R. Valencia-García, A. Rodríguez-González, G. Alor-Hernández, and J. J. Samper-Zapater, “RecomMetz: A context-aware knowledge-based

mobile recommender system for movie showtimes," *Expert Syst. Appl.*, vol. 42, no. 3, pp. 1202–1222, Feb. 2015.
7. M. A. Paredes-Valverde, R. Valencia-García, M. Á. Rodríguez-García, R. Colomo-Palacios, and G. Alor-Hernández, "A semantic-based approach for querying linked data using natural language," *J. Inf. Sci.*, p. 0165551515616311, Nov. 2015.
8. M. del P. Salas-Zárate, R. Valencia-García, A. Ruiz-Martínez, and R. Colomo-Palacios, "Feature-based opinion mining in financial news: An ontology-driven approach," *J. Inf. Sci.*, p. 0165551516645528, May 2016.
9. M. Á. Rodríguez-García, R. Valencia-García, F. García-Sánchez, and J. J. Samper-Zapater, "Ontology-based annotation and retrieval of services in the cloud," *Knowl.-Based Syst.*, vol. 56, pp. 15–25, enero 2014.
10. A. Rodríguez-González, J. E. Labra-Gayo, R. Colomo-Palacios, M. A. Mayer, J. M. Gómez-Berbís, and A. García-Crespo, "SeDeLo: Using Semantics and Description Logics to Support Aided Clinical Diagnosis," *J. Med. Syst.*, vol. 36, no. 4, pp. 2471–2481, Aug. 2012.
11. F. Baader, I. Horrocks, and U. Sattler, "Description logics," *Found. Artif. Intell.*, vol. 3, pp. 135–179, 2008.
12. E. Sanchez *et al.*, "A knowledge-based clinical decision support system for the diagnosis of Alzheimer disease," in *e-Health Networking Applications and Services (Healthcom), 2011 13th IEEE International Conference on*, 2011, pp. 351–357.
13. C. Toro *et al.*, "Using Set of Experience Knowledge Structure to Extend a Rule Set of Clinical Decision Support System for Alzheimer's Disease Diagnosis," *Cybern. Syst.*, vol. 43, no. 2, pp. 81–95, Feb. 2012.
14. R.-C. Chen, Y.-H. Huang, C.-T. Bau, and S.-M. Chen, "A recommendation system based on domain ontology and SWRL for anti-diabetic drugs selection," *Expert Syst. Appl.*, vol. 39, no. 4, pp. 3995–4006, Mar. 2012.
15. I. Horrocks *et al.*, "SWRL: A semantic web rule language combining OWL and RuleML," *W3C Memb. Submiss.*, vol. 21, p. 79, 2004.
16. P. Delir Haghighi, F. Burstein, A. Zaslavsky, and P. Arbon, "Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings," *Decis. Support Syst.*, vol. 54, no. 2, pp. 1192–1204, Jan. 2013.
17. L. Ma, H. Yu, G. Chen, L. Cao, and Y. Zhao, "Research on Construction and SWRL Reasoning of Ontology of Maize Diseases," in *Computer and Computing Technologies in Agriculture VI*, 2012, pp. 386–393.
18. M. O'Connor and A. Das, "SQWRL: a query language for OWL," in *Proceedings of the 6th International Conference on OWL: Experiences and Directions-Volume 529*, 2009, pp. 208–215.
19. E. Friedman-Hill, "Jess, the expert system shell for the java platform," *USA Distrib. Comput. Syst.*, 2002.
20. A. R. Iglesias, M. E. Aranguren, A. R. Gonzalez, and M. D. Wilkinson, "Plant Pathogen Interactions Ontology (PPIO)," *Proc. IWBBIO 2013 Int. Work-Conf. Bioinforma. Biomed. Eng. 2013*, pp. 695–702, 2013.
21. A. Halabi, "Ontology for Plant Protection," *Ontology for Plant Protection*, 2015. [Online]. Available: https://sites.google.com/site/ppontology/. [Accessed: 17-Dec-2016].
22. J. K. Patil and R. Kumar, "Advances in image processing for detection of plant diseases," *J. Adv. Bioinforma. Appl. Res.*, vol. 2, no. 2, pp. 135–141, 2011.
23. T. Rattanasawad, K. R. Saikaew, M. Buranarach, and T. Supnithi, "A review and comparison of rule languages and rule-based inference engines for the Semantic Web," in *2013 International Computer Science and Engineering Conference (ICSEC)*, 2013, pp. 1–6.

24. E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz, "Pellet: A practical owl-dl reasoner," *Web Semant. Sci. Serv. Agents World Wide Web*, vol. 5, no. 2, pp. 51–53, 2007.
25. J. M. Gómez-Pérez and C. Ruiz, "Ontological Engineering and the Semantic Web," in *Advanced Techniques in Web Intelligence - I*, J. D. Velásquez and L. C. Jain, Eds. Springer Berlin Heidelberg, 2010, pp. 191–224.
26. S. J. Clarke and P. Willett, "Estimating the recall performance of Web search engines," in *Aslib Proceedings*, 1997, vol. 49, pp. 184–189.
27. Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 42–49.
28. A. Rodríguez-González, J. Torres-Niño, R. Valencia-Garcia, M. A. Mayer, and G. Alor-Hernandez, "Using experts feedback in clinical case resolution and arbitration as accuracy diagnosis methodology," *Comput. Biol. Med.*, vol. 43, no. 8, pp. 975–986, Sep. 2013.

A Simulator's Specifications for Studying Students' Engagement in a Classroom

Latha Subramainan[1], Moamin A Mahmoud[1], Mohd Sharifuddin Ahmad[1],

Mohd Zaliman Mohd Yusoff[2]

[1] College of Computer Science and Information Technology,
Universiti Tenaga Nasional, Kajang, Selangor, Malaysia.
[2] Business Development Unit, TNB Integrated Learning Solution Sdn Bhd.

latha0522@gmail.com,{moamin, sharif}@uniten.edu.my,
zaliman.yusoff@tnb.com.my

Abstract. In this paper, we highlight the issues of poor students' engagement in classrooms and identify the attributes for the environmental settings of a proposed simulator to study the problem of students' poor engagement from the students' emotional demotion using agent-based social simulation concepts. The environmental settings of the simulation is classified into environmental factors and emotional factors. The environmental factors consist of a number of students, class session, class duration, type of subject, and year of study, while the emotional factors include the negative emotional states of student (e.g. anger, anxiety or boredom) and the emotional states of lecturers. In this simulation, a lecturer, who might have ideas on new strategies based on their experience, is able to insert a new strategy using a proposed Strategy Specification Settings Interface.

Keywords: Students' Engagement; Emotional Engagement; Agent-Based Emotions; Agent-Based Social Simulator

1 Introduction

Researchers focus on students' engagement in classrooms as a key to address issues such as low academic performance, high dropout rates, boredom and disaffection [1]. The students' engagement concept is significantly useful in predicting students' academic performance [2]. Researchers use numerous indicators to measure engagement that includes self-report, attendance rates, teacher ratings, interviews, observations, cross-cultural data and assessments grades [3, 22, 23]. However, the affective disposition of lecturers and students has not been comprehensively studied to understand its effects on students' engagement in classrooms.

In this paper, we aim to identify the symptoms of students' negative emotional states and a possible strategy to improve the students' engagement in a classroom based on the negative emotional states of anger, anxiety and boredom. Previous

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_25

researchers aim to improve students' engagement randomly and not limited to cater any specific emotions. We also believe lecturers in a class are always able to dynamically change their teaching styles or strategies based on emotional feedback portrayed by students [7].

This paper is an extension to our work in studying and enhancing students' engagement strategies using an agent-based emotion model [8, 21]. In our earlier research, we presented a conceptual agent-based emotion model animated by an agent-based social simulation [Ref]. We use an agent platform for animating a classroom environment in running multiple settings and obtaining results that are more accurate. The rudimentary theory of this study is based on four main elements; the selected strategy to control engagement, the engagement level of students, the emotional state of a lecturer and the emotional states of students in a classroom. Hence, in this paper we highlight factors influencing lecturers' emotions and students' negative emotional states that impact on the students' engagement in a classroom. We also discuss the potential factors influencing strategies to improve student engagement.

2 Related Works

Students' engagement serves as a protective factor against students dropout and involvement in perilous activities [9]. But students' emotional engagement has received little attention due to its lack of conceptual clarity [4, 9]. According to Kort et al. [10], "students who are anxious, angry, or depressed don't learn; people who are caught in these states do not take in information efficiently or deal with it well". Therefore, the importance of emotion in classroom teaching cannot and should not be taken lightly. Pekrun et al. [11], identify a variety of emotions experienced by lecturers and students, including joy, enthusiasm, hope, relief, pride, gratitude, admiration, sadness, anger, anxiety, hopelessness, shame and guilt, disappointment, boredom, envy, disrespect, and surprise.

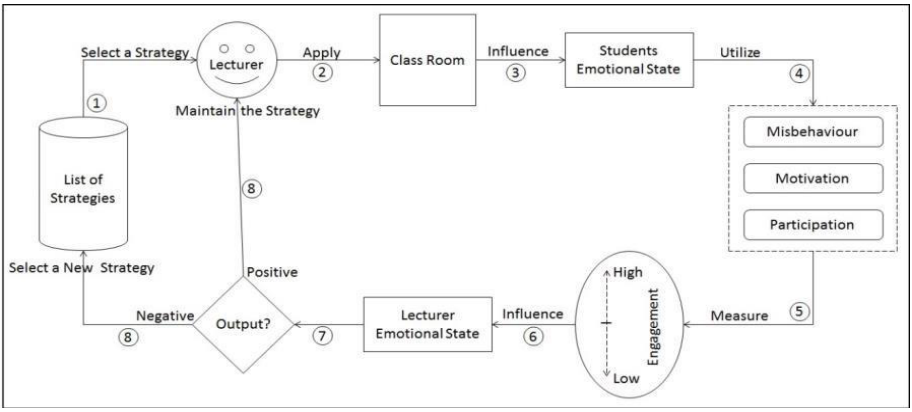
A lecturer could monitor a student's engagement level in a classroom via indirect indicators, for instance, the intensity of participation in classroom discussions, attendance, commitment to a given task, time spent on assessments, intensity of concentration during ongoing lessons and the motivation or interest shown on a particular course materials [20]. A recent study has shown that various factors could invoke lecturers' emotions depending on the context which the lecturer encounters, for instance, specific class, subject domain, and specific lesson [12]. However, lecturers usually report that students' behaviors, discipline and their interpersonal relations to their students as significant sources of their emotions. Studies suggest that enjoyment is the most prominent positive emotion and anger is the most frequently experienced negative emotion, lecturers experience while teaching [12]. Lecturers' emotions subsequently influence their instructional behaviors during lessons which could then impact on student academic outcomes and behaviors. Lack of engagement also indicates lecturers' emotions [11]. A recent research reports that lecturers' anger is triggered particularly if the "students' misbehaved or did not engage in the learning

process" [13]. Hagenauer et al. [13], argue that students' misbehavior in the classroom represents a threat to their instructional or management goals which is strongly related to negative emotions. Students' misbehavior, motivation and participation levels have been shown to be an important cause for negative emotional experiences in lecturers and one of the main sources for lecturers exhaustion and stress levels [14, 15]. Studies have indicated emotions are always involved in the learning process. Normally, students feel proud or pride about scoring good grades in assessments, anxious or frustrated when they are not able to understand the course materials, and may get angry with a lecturer who is uncaring and being unfair to them. They might also feel bored learning a topic which disinterest them. Anger and boredom are reported as occurring fairly frequently in a classroom [11]. In addition, boredom is reflected during lessons, anxiety and anger are usually experienced as fear of failure or anger at failure [16]. Both lecturers and students experience a total of six emotions in the classroom. They share a few common emotions such as enjoyment, pride, anger, anxiety and boredom in the classroom. However, in our study, we focus on negative emotions of students, which are anger, anxiety and boredom. These emotions selection is based on two reasons: (a) it occurs frequently in a classroom [10] and (b) it is based on the control-value theory of achievement [14]. Only negative emotions can trigger a lecturer to change their teaching strategy to improve students' emotional state which then subsequently improve student engagement in a classroom. If positive emotion such as enjoyment occurs among the student and keeps the student engaged throughout the lesson, then the lecturer can maintain the strategy.

An agent-based social simulation (ABSS) is a method to model systems that comprise of individual, autonomous, cooperating agents [12]. This method can be used to model human behaviors and their effects to others. This method has enjoyed widespread use in emergency evacuations, car pooling and disaster response. One way of characterizing the research area of Agent-Based Social Simulation (ABSS) is that it constitutes the intersection of three scientific fields, namely, agent-based computing, the social sciences, and computer simulation. ABSS is an ideal approach to simulate systems that comprise of autonomous and interacting agents. Generally, ABSS is used to examine a phenomenon and solve problems that are challenging for humans to study and provide solutions. It can be used in a variety of areas (e.g., computer games) to represent human societies. Moreover, artificial and natural occurrences can be represented in the simulation. The reason for doing computer simulations is usually to gain a deeper understanding of the phenomenon. Research have proven that ABSS is a powerful tool for modelling and understanding phenomena in various areas such as economics and trading, health care, urban planning and social events. However, according to recent research, even though autonomous agents have been popular for decades, they are still in the early phases of implementations [3].

3 A Proposed Agent-Based Emotion Framework

Intelligent software agents have been widely used in distributed artificial intelligence and due to their autonomous, self-interested, rational abilities [24, 25, 26, 27], and social abilities [28, 29, 30]. The preliminary framework of this proposal is based on four main elements in a classroom; the selected strategy to control engagement, the engagement level of students, the emotional state of a lecturer and the emotional states of students. The process starts by selecting a strategy and applying it to the students, which influences the students' emotional state. By



analyzing three variables, students' misbehaviors, motivation and participation, the engagement level of students can be measured. The result of engagement measurement positively or negatively influences the lecturer's emotion. If negatively, the lecturer change to another strategy that would trigger the students' emotion and eventually improve engagement. The lecturer again measures the engagement level and decide whether to apply another strategy or maintain the existing one. Figure 1 below illustrates the framework.

Fig.1. The Proposed Framework [8]

As shown in Figure 1, the Lecturer Agent (AL) selects (1) a strategy from the list of strategies stored in the tool. The AL then applies (2) the strategy during classroom session which eventually influences (3) Students' Agents (ASTs) emotional state. By utilizing (4) the factors misbehaviour, motivation and participation, the AL measures (5) the engagement level. The level of engagement influences (6) the AL emotional state either positively or negatively (7). Negative emotion triggers (8) the AL to select a new strategy which subsequently influences the ASTs emotional states that would improve their engagement in the classroom. The Lecturer maintains (8) the strategy if positive emotion occurred. The process continues until the ASTs displays positive factors of behavior, motivation and participation. In this paper, we identify the factors that influence students' engagement where AL is able to propose the most appropriate strategy to improve students' negative emotional state and subsequently improve students' engagement in a classroom.

4 Student Emotional Engagement Evaluation

Changes in the emotional state of a student influences the engagement level which then subsequently effect the emotional states of lecturer. For example, whenever students get bored during the lesson, they would start playing games, talking to each other, and cease to pay attention to the lesson, resulting in a drop in their engagement to class lesson. Consequently, we found that students' emotional engagement could be inferred via observing the factors of misbehavior, motivation and participation.

Table 1 shows students' engagement evaluation with possible indications of each engagement factor.

Table 1. Student Emotional Engagement Evaluation

Student Engagement Factors	Indications	Possible Emotional states of Students	Engagement Level	Possible Emotional
Misbehavior [12]	• Striking out • Verbally or physically aggressive • Spoken in a raised voice • Acting in abusive manner	Anger	Low	Anger
	• Playing video games • Get away from class	Boredom	Low	Anxiety
Motivation [15]	• Non-attentiveness • Daydreaming • Talking out of turn	Boredom	Low	Anxiety
Participation [32]	• Avoidance or refusal to participate • Inability to initiate conversations • Avoiding eye contact	Anxiety	Low	Boredom

In a qualitative study, teachers revealed that their emotions are influenced by various aspects include students' misbehavior and poor relationships with them [12]. Besides, highly motivated students are related to teachers' enjoyment and vice versa [15]. Teachers also prefer to work with students who achieve their success through effort and participation [32]. Apparently, low engagement level negatively influences the emotional state of a lecturer. Therefore, the lecturer needs to initiate a strategy to improve students' engagement in a classroom. A good strategy is important for a teaching process. According to the literature, common components that forms a strategy are students' interest, teaching materials, time and resources management, teachers' planning skills, teacher enthusiasm and classroom environment (e.g. number of students, time of day, duration of class, age and ability of the students) [16, 17, 18].

In our study, we emphasize on the class environment factors, such as the number of students (30 or 60 or 90). Size of a classroom affects the motivation of students to engage verbally in classroom [33]. For instance, if a full classroom of 60 students, then the students in the back would not be able to listen the lesson properly. Consequently, it cause boredom among the students and they would start misbehave and pay no attention to lesson. Other strategies would be suitable for small number of students. Some factors includes class session (e.g. morning, afternoon/evening), class duration (1 hour, 2 hours or more).

Type of subject (e.g. theoretical or conceptual) also influence a strategy. A study revealed, student participation in conceptual class is high when a teacher divide students into three to five in a group and delegating the work [34]. In [35], study has revealed that quality of effort increases as the year of study increases. Therefore, year of study (first year or third year) plays a role in influencing a strategy. Furthermore, emotional factors such as negative emotional states of students (e.g. anger, anxiety or boredom) also occur based on the effectiveness of a strategy. According to [36], students seeing that a teacher's teaching strategy mostly as the main source of their boredom. Only small number of students admit the reasons for their misbehavior. To sum up, we develop the simulator based on environment and emotional factors.

5 The Simulation Functions

The proposed simulator could be utilized by researchers and educationists to investigate the problem of students' engagement. A lecturer needs to key-in information of a class Environment Settings. The required information are Subject Name, No. of Students, Class Session, Class Duration, Type of Subject, Year of Study and finally estimate the percentage of misbehavior, motivation and participation levels of students based on their last class session experience which eventually predict the students' emotional states . Next, the lecturer can select any initial strategy and test it for multiple times to observe the performance of students' engagement level.

A lecturer, who might have ideas on new strategies based on their teaching experience, is able to insert a new strategy using the Strategy Specification Settings Interface and use the Likert scale to appropriately assign the likeability of the strategy. To run the simulator, a lecturer/researcher/educationist needs to insert information of a particular class environment and the percentage of misbehavior, motivation and participation level based on observations on students' behavior from previous sessions, and select an initial strategy. Having run the simulator, the software agents captures the environment setting data and play scenario to test the effectiveness of the selected strategy. The successful rate of the strategy will be shown in percentages. For example, Positive Peer Review strategy effectiveness might be 40% suit for particular environment settings. The agent is also able to recommend other strategies that would netter suit the given environment setting.

6 Conclusion and Future Work

In this paper, we present the development of our work-in-progress about constructing a virtual classroom environment to study and enhance students' engagement strategies using an agent-based emotion model. We explore the environmental settings of students' emotional demotion for a proposed simulator investigating the problem of the students' poor engagement. We first presented an agent-based emotion framework to compute students' emotions towards their lecturer and to recommend the best possible strategy based on the negative emotions of students.

The environmental settings of the simulation is classified into the environmental and emotional factors based on the scope of this research. Environmental factors such as the number of students, class session, class duration, type of subject, year of study, and emotional factors such as negative emotional states of student (e.g. Anger, Anxiety or Boredom).In the future work, we shall create a virtual environment simulating a classroom dynamics based on the settings. Experts from the field of computer science and social studies will validate the simulation.

7 References

1. Fredricks, J. A., Filsecker, M., & Lawson, M. A. (2016). Student engagement, context, and adjustment: Addressing definitional, measurement, and methodological issues.
2. Salleh, A. M., Desa, M. M., & Tuit, R. M. (2013). The Relationship between the Learning Ecology System and Students' Engagement: A Case Study in Selangor. *Asian Social Science*, *9*(12), 110.
3. Fredricks, J. A., & McColskey, W. (2012). The measurement of student engagement: A comparative analysis of various methods and student self-report instruments. In *Handbook of research on student engagement* (pp. 763-782). Springer US.
4. Pekrun, R., Cusack, A., Murayama, K., Elliot, A. J., & Thomas, K. (2014). The power of anticipated feedback: Effects on students' achievement goals and achievement emotions. *Learning and Instruction*, *29*, 115-124.
5. Kazmi, A. (2010). Sleepwalking through Undergrad: Using Student Engagement as an Institutional Alarm Clock. *College Quarterly*, *13*(1), n1.
6. Castro, S., Granlund, M., & Almqvist, L. (2015). The relationship between classroom quality-related variables and engagement levels in Swedish preschool classrooms: a longitudinal study. *European Early Childhood Education Research Journal*, 1-14.
7. Kung-Keat, T., & Ng, J. (2016). Confused, bored, excited? An emotion based approach to the design of online learning systems. In *7th International Conference on University Learning and Teaching (InCULT 2014) Proceedings* (pp. 221-233). Springer Singapore.
8. Subramainan, L., Mahmoud, M. A., Ahmad, M. S., & Yusoff, M. Z. M. (2016, August). A conceptual emotion-based model to improve students engagement in a classroom using agent-based social simulation. In *User Science and Engineering (i-USEr), 2016 4th International Conference on* (pp. 149-154). IEEE.
9. Skinner, E., Furrer, C., Marchand, G., & Kindermann, T. (2008). Engagement and disaffection in the classroom: Part of a larger motivational dynamic?. *Journal of Educational Psychology*, *100*(4), 765.

10. Kort, B., Reilly, R., & Picard, R. W. (2001). An affective model of interplay between emotions and learning: Reengineering educational pedagogy-building a learning companion. In *Advanced Learning Technologies, 2001. Proceedings. IEEE International Conference on* (pp. 43-46). IEEE.
11. Pekrun, R., Goetz, T., Frenzel, A. C., Barchfeld, P., & Perry, R. P. (2011). Measuring emotions in students' learning and performance: The Achievement Emotions Questionnaire (AEQ). *Contemporary educational psychology*, *36*(1), 36-48.
12. Frenzel, A. C., Becker-Kurz, B., Pekrun, R., & Goetz, T. (2015). Teaching this class drives me nuts!-Examining the person and context specificity of teacher emotions. *PloS one*, *10*(6), e0129630.
13. Hagenauer, G., Hascher, T., & Volet, S. E. (2015). Teacher emotions in the classroom: associations with students' engagement, classroom discipline and the interpersonal teacher-student relationship. *European journal of psychology of education*, *30*(4), 385-403.
14. Pekrun, R., & Linnenbrink-Garcia, L. (2014). Introduction to emotions in education. *International handbook of emotions in education*, 1-10.
15. Goetz, T., Frenzel, A. C., Pekrun, R., Hall, N. C., & Lüdtke, O. (2007). Between-and within-domain relations of students' academic emotions. *Journal of Educational Psychology*, *99*(4), 715.
16. Tsouloupas, C. N., Carson, R. L., Matthews, R., Grawitch, M. J., & Barber, L. K. (2010). Exploring the association between teachers' perceived student misbehaviour and emotional exhaustion: The importance of teacher efficacy beliefs and emotion regulation. *Educational Psychology*, *30*(2), 173-189.
17. Papa-Gusho, L., & Biçaku-Çekrezi, R. (2015). Factors that Affect Effective Planning Skills of the Teacher in the Classrooms. *Academic Journal of Interdisciplinary Studies*, *4*(3 S1), 560.
18. Bilash , O. (2009, May). Lesson Planning in the Language Classroom. Retrieved September 10, 2016, from http://sites.educ.ualberta.ca/staff/olenka.bilash-/Best%20of%20Bilash/lessonplanning.html
19. Making Effective Instructional Choices . (n.d.). Chapter (7), 133-145. Retrieved September 8, 2016, from http://teachingasleadership.org/sites/default/files/Related-Readings/IPD_Ch7_2011.pdf
20. Newmann, F. M. (1992). *Student engagement and achievement in American secondary schools*. Teachers College Press, 1234 Amsterdam Avenue, New York, NY 10027 (paperback: ISBN-0-8077-3182-X, $17.95; hardcover: ISBN-0-8077-3183-8).
21. Subramainan, L., Mahmoud, M. A., Ahmad, M. S., & Yusoff, M. Z. M. (2016). An Emotion-based Model for Improving Students' Engagement using Agent-based Social Simulator. *International Journal on Advanced Science, Engineering and Information Technology*, *6*(6).
22. Subramainan, L., Mahmoud, M. A., Ahmad, M. S., & Yusoff, M. Z. M. (2016, August). Evaluating students engagement in classrooms using agent-based social simulation. In *Agent, Multi-Agent Systems and Robotics (ISAMSR), 2016 2nd International Symposium on* (pp. 34-39). IEEE.
23. Subramainan, L., Yusoff, M. Z. M., & Mahmoud, M. A. (2015, August). A classification of emotions study in software agent and robotics applications research. In *Agents, Multi-Agent Systems and Robotics (ISAMSR), 2015 International Symposium on* (pp. 41-46). IEEE.
24. Mahmoud, M. A., & Ahmad, M. S. (2015, August). A self-adaptive customer-oriented framework for intelligent strategic marketing: A multi-agent system approach to website

development for learning institutions. In Agents, Multi-Agent Systems and Robotics (ISAMSR), 2015 International Symposium on (pp. 1-5). IEEE.
25. Jassim, O. A., Mahmoud, M. A., & Ahmad, M. S. (2015). A Multi-agent Framework for Research Supervision Management. In Distributed Computing and Artificial Intelligence, 12th International Conference (pp. 129-136). Springer International Publishing.
26. Mahmoud, M. A., & Ahmad, M. S. (2016, August). A prototype for context identification of scientific papers via agent-based text mining. In Agent, Multi-Agent Systems and Robotics (ISAMSR), 2016 2nd International Symposium on (pp. 40-44). IEEE.
27. Mahmoud, M. A., & Ahmad, M. S. (2015, August). A self-adaptive customer-oriented framework for intelligent strategic marketing: A multi-agent system approach to website development for learning institutions. In Agents, Multi-Agent Systems and Robotics (ISAMSR), 2015 International Symposium on (pp. 1-5). IEEE.
28. Moamin, M. A. H. M. O. U. D., Sharifuddin, A. M., & Zaliman, M. Y. M. (2016). Development and implementation of a technique for norms-adaptable agents in open multi-agent communities. *Journal of Systems Science and Complexity*, *29*(6), 1519-1537.
29. Mahmoud, M. A., Ahmad, M. S., & Yusoff, M. Z. M. (2016, March). A Norm Assimilation Approach for Multi-agent Systems in Heterogeneous Communities. In Asian Conference on Intelligent Information and Database Systems (pp. 354-363). Springer Berlin Heidelberg.
30. Mahmoud, M. A., Mustapha, A., Ahmad, M. S., Ahmad, A., Yusoff, M. Z. M., & Hamid, N. H. A. (2013). Potential norms detection in social agent societies. In Distributed Computing and Artificial Intelligence (pp. 419-428). Springer International Publishing.
31. Kravari, K., & Bassiliades, N. (2015). A survey of agent platforms. Journal of Artificial Societies and Social Simulation, 18(1), 11.
32. Biddle, S., & Goudas, M. (1997). Effort is virtuous: Teacher preferences of pupil effort, ability and grading in physical education. *Educational Research*, *39*(3), 350-355.
33. Abdullah, M. Y., Bakar, N. R. A., & Mahbob, M. H. (2012). Student's Participation in Classroom: What Motivates them to Speak up?. *Procedia-Social and Behavioral Sciences*, *51*, 516-522.
34. Kamarudin, N., Halim, L., Osman, K., & Meerah, T. S. M. (2009). Pengurusan penglibatan pelajar dalam amali sains. *Jurnal Pendidikan Malaysia*, *34*(1), 205-217.
35. Pace, C. R. (1990). The Undergraduates: A Report of Their Activities and Progress in College in the 1980's.
36. Stuchlíková, I., Gläser-Zikuda, M., & Janík, T. (2013). Emotional Aspects of Learning and Teaching: Reviewing the Field− Discussing the Issues. *Orbis scholae*, (2), 7-22.

Energy Analyzer Emulation for Energy Management Simulators

Luis Gomes[1], and Zita Vale[1]

[1]GECAD – Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development, Institute of Engineering – Polytechnic of Porto (ISEP/IPP), Rua Dr. António Bernardino de Almeida, 431, 4200-072 Porto, Portugal

{lufog,zav}@isep.ipp.pt

Abstract. The simulation of microgrids to testing and validate energy management methodologies are an important step to take before the massive implementation of microgrids. However, microgrids are usually unavailable for R&D centers to perform tests and validations. To solve this issue is important to get the simulations closer to the reality, using real energy analyzers and loads. However, again, R&D centers lack from funding and space to buy and mount several loads in their laboratories. To solve this issue, this paper proposes a multi-agent system simulator for microgrids and an energy analyzer emulator that can be used to emulate individual loads or entire houses, and therefore, bringing the pure simulation closer to the reality.

Keywords: Energy analyzer emulator · Forecasting · Load emulation · Multi-agent system

1 Introduction

The power systems have been changing their paradigm from a centralized system to a decentralized system [1]. This has impact in the generation as well has in the energy management. The appearance of a smart grids with microgrids and smart homes are, in present day, becoming a reality, with more and more applications worldwide [2,3].

The application of microgrids in power system brings advantages for the majority of players [4]. However, to achieve the massive implementation of microgrids some steps are needed. There are a need of capable simulators that can represent microgrids in a way that energy management methodologies can be tested and validated.

The present work has been developed under the EUREKA - ITEA2 Project M2MGrids (ITEA-13011), Project SIMOCE (ANI|P2020 17690), and has received funding from FEDER Funds through COMPETE program and from National Funds through FCT under the project UID/EEA/00760/2013 and SFRH/BD/109248/2015.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_26

The use of a Multi-Agent System (MAS) to analyze and improve microgrids behavior, regarding the energy management, is a perfect fit [5,6]. The use of MAS allows the representation microgrid players, where we have individual players cooperating with each other (trying to combine their individual goals with the group goals).

The consumption forecasting is an important part of the energy management systems. Usually, in energy management system, we work in future periods [7], like day-ahead or hour-ahead. It is only possible to manage energy for a future period if we know what will happen, or at least have an idea about how the things will unfold. Forecasting algorithms are necessary for in advance energy management solutions. And therefore, there are several works concerning this issue [8,9].

Having the simulations and the methodologies, is necessary to go further, close to the real implementation. This is a gap that most of works do not achieve. This paper proposes an energy analyzer emulator that can be integrated in the simulations. The emulator, named Virtual to Reality (V2R), works as a real analyzer and presents to the system as one. This enables the system to test and work with, what it seems, a real analyzer. The use of emulators brings advantages compared to the use of real analyzers. With real analyzers, the R&D center must buy real controllable loads and is not a scalable solution if we think in a simulation with 100 houses. Applying emulators, and going closer to reality is possible to have better simulations than the pure simulations existing today that do not deal with hardware [10,11], standing themselves far from real implementations in pilots and/or microgrids.

The main contribution of this paper in the integration of V2R in a microgrid management multi-agent system. This integration will enable the efficient and realistic simulation of microgrids in the multi-agent system. The paper presents the V2R and the multi-agent system, which started as a simulator and now is running with real buildings. The case study presents a consumption forecast of the microgrid, proving V2R capabilities of working as an energy analyzer emulator.

After this introduction section, the paper will describe the multi-agent system used for the microgrid implementation in section 2. A brief explanation of forecasting is given in section 3. Section 4 presents the proposed energy analyzer emulator. The case study is presented in section 5. And the main conclusions are presented in section 6.

2 Multi-Agent System for Microgrid Implementation

This paper will use the Multi-Agent Smart Grid Platform (MASGriP) to simulate a microgrid. MASGriP is a complete platform used for studies along the years [12]. It was developed in Java and can combine simulated agents (representing simulated buildings) with real agents (representing real buildings).

In order to test a microgrid, our R&D center implemented a microgrid in our buildings [13][14]. Using MASGriP is possible to represent each of the microgrid players in a computational platform. MASGriP was built using JADE framework, each player is independent from the each other and work together using a cooperation approach. The communication between agents are compliant to FIPA specifications. Fig. 1 shows the implementation used in this paper:

- ***Microgrid player*** – responsible for microgrid management, all the other players responds to him. The role of this agent is to manage the microgrid in benefit of their players, focusing on the group goals to increase the energy quality. This agent is a virtual agent, not representing a physical building, the location of this agent is in the microgrid operation room (available in GECAD [13]);
- ***Building I*** – responsible for Building I of ISEP (where GECAD is based). The agent monitors the energy using an energy analyzer with Ethernet connection. The agent stores the data, in a SQL Server, and represents the building in the microgrid;
- ***Building N*** – responsible for the Building N of ISEP where there is a Programmable Logic Controller (PLC) and a ZigBee gateway that monitors 5 energy analyzers interconnected through RS-485. Building N has 9 controllable air conditioners. The agent monitors, stores, in a SQL Server, and controls the building's energy;
- ***Simulated Building*** – this agent was integrated in the simulation for the paper use case. This agent will integrate energy analyzers emulators and store, in a SQL Server database, their data. The agent will then represent the energy analyzers emulators in the microgrid. Besides its name, the agent works in MAS as a real building agent, the agent do not know that its energy analyzers are emulated.

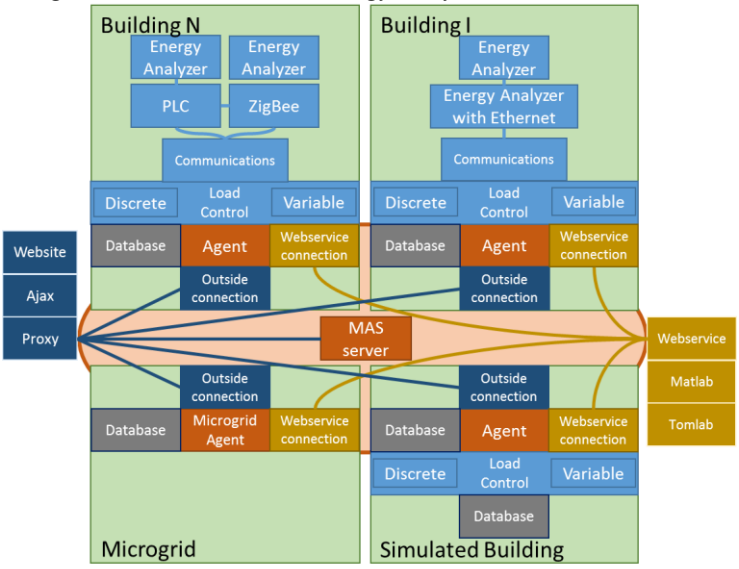


Fig. 1. MASGriP GECAD implementation

The communication between agents and energy analyzers, installed in the buildings, are done through Modbus/TCP, direct to the energy analyzers or using a PLC in the middle. The agent configuration if done using a XML file where is possible to specify the energy analyzers of the agent, agent name and MAS name and location. The Simulated Building agent and microgrid agent, do not have the knowledge that some loads are emulated. They work thinking that they have real loads installed in real buildings.

MASGriP includes algorithms needed for energy management, such as, forecasting algorithms [8] and scheduling algorithms [15]. The algorithms are available for all

agents. The algorithms are open to the agents in a web service in order to improve the execution time and make all the algorithms available for all the agents. This web service supports the programming languages of: C; C++; C#; R; Matlab; and Tomlab.

MASGriP agents are accessible using a webpage. All the agents implements a simple API using JSON messages that provides monitoring and control requests. The webpage allows the users to visualize real-time and stored data. The microgrid agent webpage also allows the execution of energy management actions in the microgrid.

This implementation not use generation units in any agent. However, MASGriP is able to integrate generation. By default, microgrid agent asks the consumption value each second. But agents can refresh (monitor) their values (energy analyzers) using other time periods. For instance, Building N is using a time period of 10 seconds. A more detailed description of MASGriP implementation is present in [13, 8,14].

3 Consumption Forecasting

The forecasting algorithms are available to MASGriP agents using the platform web service. In there are two types of forecasting algorithms: Artificial Neural Networks (ANN); and Support Vector Machine (SVM). Both algorithm were created to learn from data and provide solutions according to what they learned. In MASGriP they are used to learn profiles of consumption, or generation, and then provide forecasts. Because they are algorithms based on learning models, they are highly dependent on the learn data [8]. For this reason, the inputs to use should be tested.

The ANN can be seen in [8] where a detailed analysis of ANN is done. In this paper the forecasting algorithm used is the SVM. The SVM available in the web services allows the specification of the kernel to use: Linear; Polynomial; Radial; and Sigmoid. The SVM algorithm tries to find a regression between previous data to predict/forecast new data.

In the MASGriP implementation of these paper it will be executed a forecasting for the next 24 hours of consumption in each consumer agent (Fig. 2). Is defined as a consumer agent, agents that have energy consumption loads (Building N, Building I and Simulated Building).

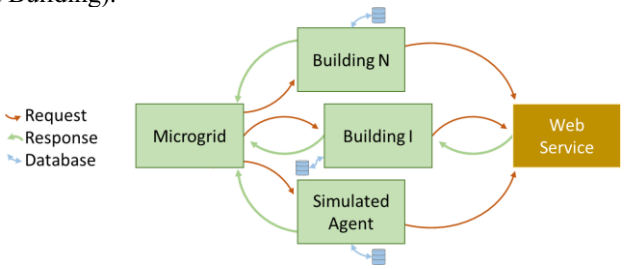


Fig. 2. Process for microgrid forecasting

A microgrid forecasting can be done using microgrid agent. Fig. 2 shows this process. The microgrid agent will individually, and simultaneous, request each consumer agent to provide a forecast for a given ahead period. At this moment, each

agent/player should execute the forecast it wants (ANN or SVM). The agent uses the stored data to feed the input of the forecasting and then waits for the web service response. The forecasting are executed locally and in a distributed way throughout the MAS. After processing the forecast, the agents send the data back to the microgrid agent. Then, microgrid agent, will aggregate the forecasts and create a global forecast. This process requires more process in the web service side, however, it maintains the consumptions data in the consumer side (no stored data goes to the microgrid agent).

4 V2R - Energy Analyzer Emulator

Energy analyzers are a useful tool for energy monitoring in new and old buildings. These products can be found on the market with a widely range of prices. There are single-phase and three-phase. Some of them also allows the on/off control of an output port. They can be mounted in the entire building, a single room or a single load.

The capability of measuring energy is important for power systems R&D centers that have a need for build scenarios with real data. The protocol normally used for communications is Modbus. This is a simple request-response protocol that works in two communication standards: RS-485; and TCP/IP. Modbus/TCP is used through TCP/IP using Ethernet, in order to use this protocol the energy analyzer need an Ethernet connection or is needed a RS-485 to TCP/IP interface. Modbus/RTU is used through RS-485 and is adopted by the majority of manufactures.

The energy analyzers work perfectly if the goal is to measure real loads with real consumption. However, is not possible to use them to simulate loads. This is a problem for implemented and installed solutions that need to perform tests and validation of various scenarios. To solve this issue, this paper proposes the use of Virtual to Reality (V2R) emulator [16]. This is an emulator built to overpass the limitations of the energy analyzers. V2R allows the simulation of loads using the installed infrastructure. V2R is basically an energy analyzer for load emulation, which uses the same Modbus/RTU communication as an energy analyzer. Fig. 3 shows the V2R components.

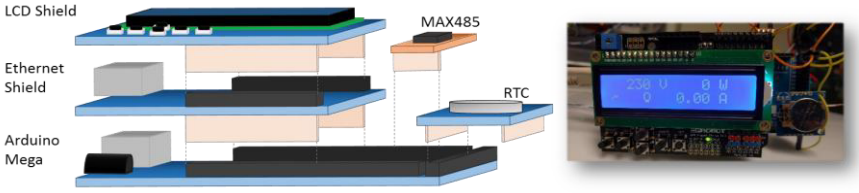


Fig. 3. V2R emulator (physical components)

V2R provides a LCD display to monitor and configure some parameters (Fig. 3). The emulator allows six types of emulations:

- ***Chart Load*** – this type of emulation follows the data in a chart using the Real-Time Clock (RTC) to mimic the input consumption using the data stamp. This is an uncontrolled load that follows a profile consumption, it is a good option for a refrigerator emulation without control;

- ***Discrete Load*** – this is a controllable load (using the LCD shield buttons), the configuration defines the on state consumption and the off state consumption. Them the emulator turns the load on and off according to the pressing of the button (by the user or using a webpage interface). This can be indicated for heater;
- ***Variable Load*** – similar to the discrete load but the load is dimmable, this is a good choice for a lamp;
- ***Contextual Load*** – this emulation is the combination of the chart load with the discrete load. This load is controllable but follows a consumption profile when it is on. For instance, this emulation can be used for a washing machine, that when is turned on it will follow the consumption profile of a washing program;
- ***OPAL-RT Load*** – some loads cannot be easily emulated, for this reason V2R can connects to OPAL-RT that is a real-time simulator able to simulate even the most complex loads. V2R will connect to OPAL-RT and display the consumptions simulated in OPAL-RT. This can be indicated for a wind turbine;
- ***Modbus/TCP Mirror Load*** – the mirror load enables the use of loads in a system where they are not, this type reads the consumption of a distant load (using Modbus/TCP), that can be in the other side of the world, and mirrors its' consumptions.

5 Case Study

For this case study it will be used the MASGriP platform for the multi-agent system to represent the implementation of Fig. 1. The user requests a forecast to the microgrid agent. This request starts three new requests simultaneous where the microgrid agent requests the other players (Building N, Building I and the Simulated Building) to execute a forecasting for next day consumptions.

Building N and Building I agent will use real data of the buildings. For the forecasting, it will be used the two last months for training. The forecast of one day is composed by 24 values (one value of consumption per hour). The agents will execute the forecasting using SVM algorithm, which are available in the web service and was previously discussed in Section 3.

The Simulated Building was created using V2R. The agent, representing the Simulated Building in MASGriP, don't know it's working with simulated loads, the V2R loads are integrated as real loads. The V2R emulates the loads using chart emulation that execute emulations using previous store data. The Simulated agent has three V2R connected: one that emulates a refrigerator; on for a water heater emulation; and other that represents the rest of the house. For the last V2R is used a data set available in the Intelligent Data Mining and Analysis Working Group website[1], identified with the name Private Home 6. For the Simulated Building forecasting, it is used a two month period for training, equal to the other agents.

The microgrid agent waits for the agents individual responses and then combine them to obtain the microgrid forecast. Fig. 4 shows the microgrid forecast using the

[1] http://sites.ieee.org/pes-iss

discrimination of the agents. Building N and Building I are offices with a high volume of consumption, while the Simulated Building represents a single house.

Remember, that the Simulated Building agent don't know that the loads it is reading are emulated. The results shows that the V2R worked perfectly, given the system the impression that the building was real with real energy analyzers reading real loads.

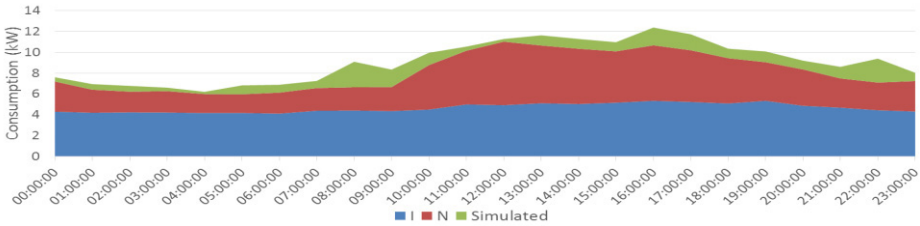


Fig. 4. Microgrid forecast by buildings

The forecasting algorithm result represents the forecast of the microgrid. The highest consumption during the day will be between 10 a.m. and 7 p.m.. In the opposite side, in the Simulated Building, because is a residential house, the highest consumption are at 8 a.m. and 10 p.m., these are the periods where people tend to have bigger consumption inside their houses. Building I, because of their characteristics, do not present significant changes of consumption through the entire day. In order to improve the forecast, new inputs must be studied and analyzed.

6 Conclusions

The need for microgrids studies is a reality, the power system paradigm demands new solutions involving distributed generation and distributed energy management systems. But in the other hand, the execution of tests and validations from the scientific community is limit to the available hardware, in very uncommon for the R&D groups to have an operational microgrid to test on.

The use of multi-agent systems and load emulation enables quick and efficient tests and validation, regarding energy management methodologies (but not in a hardware point of view). Multi-agent systems are already used for power systems, this paper proposes the use of a multi-agent system with load emulators that can be used with real systems, combining both worlds. The application of energy analyzers emulators, bring the systems closer to reality. The case study shows exactly this capability. In the case study the multi-agent system, which is implemented in real buildings, worked also with a simulated building (built with load emulators) without knowing that the building do not exist.

The combination between multi-agent systems, real buildings and emulators are an important step towards the real implementation of energy management systems in pilots' sites and new microgrids' sites.

7 References

1. Klessmann, C., Held, A., Rathmann, M., Ragwitz, M.: Status and perspectives of renewable energy policy and deployment in the European Union—What is needed to reach the 2020 targets?. In: Energy Policy, Volume 39, Issue 12, pp.7637-7657 (2011)
2. Washom, B., Dilliot, J., Weil, D., Kleissl, J. Balac, N. Torre, N., Richter, C.: Ivory Tower of Power: Microgrid Implementation at the University of California, San Diego. In: IEEE Power and Energy Magazine, vol. 11, no. 4, pp. 28-32 (2013)
3. Tsui, K.M., Chan, S.C.: Demand Response Optimization for Smart Home Scheduling Under Real-Time Pricing. In: IEEE Transactions on Smart Grid, vol. 3, pp. 1812-1821 (2012)
4. Fu, Q., Montoya, L. F., Solanki, A., Nasiri, A., Bhavaraju, V., Abdallah, T., Yu, D.C.: Microgrid Generation Capacity Design With Renewables and Energy Storage Addressing Power Quality and Surety. In: IEEE Transactions on Smart Grid, vol. 3, no. 4 (2012)
5. Dou, C., Lv, M., Zhao, T., Ji, Y., Li, H.: Decentralised coordinated control of microgrid based on multi-agent system. In: IET Generation, Transmission & Distribution, vol. 9, no. 16, pp. 2474-2484 (2015)
6. Vale, Z., Pinto, T., Praça, I., Morais, H.: MASCEM: Electricity Markets Simulation with Strategic Agents. In: IEEE Intelligent Systems, vol. 26, no. 2, pp. 9-17 (2011)
7. Silva, M., Morais, H., Sousa, T., Faria, P., Vale, Z.: Time-horizont distributed energy resources scheduling considering the integration of real-time pricing demand response. In: IEEE Eindhoven PowerTech, Eindhoven, pp. 1-6 (2015)
8. Vinagre, E., Gomes, L., Vale, Z.: Electrical Energy Consumption Forecast Using External Facility Data. In: IEEE Symposium Series on Computational Intelligence, 659-664 (2015)
9. Wan, C., Zhao, J., Song, Y., Xu, Z., Lin, J., Hu, Z.: Photovoltaic and solar power forecasting for smart grid energy management. In: CSEE Journal of Power and Energy Systems, vol. 1, no. 4, pp. 38-46 (2015)
10. Habib, H. F., Yossef, T., Cintuglu, M., Mohammed, O.: A Multi-Agent Based Technique for Fault Location, Isolation, and Service Restoration. In: IEEE Transactions on Industry Applications, no.99, pp.1-1(2017)
11. Rakesh, G., Pindoriya, N. M.: Simulation and experimental study of single phase PWM AC/DC converter for Microgrid application. In: 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), Delhi (2016)
12. Morais, H., Vale, Z., Pinto, T., Gomes, L., Fernandes, F., Oliveira, P., Ramos, C.: Multi-Agent based Smart Grid management and simulation: Situation awareness and learning in a test bed with simulated and real installations and players. In: IEEE Power & Energy Society General Meeting, Vancouver, BC, 2013, pp. 1-5 (2013)
13. Gomes, L., Silva, J., Faria, P., Vale, Z.: Microgrid demonstration gateway for players communication and load monitoring and management. In: Clemson University Power Systems Conference (PSC), Clemson, SC, 2016, pp. 1-6 (2016)
14. Gomes, L., Lefrançois, M., Faria, P., Vale, Z.: Publishing real-time microgrid consumption data on the web of Linked Data. In: 2016 Clemson University Power Systems Conference (PSC), Clemson, SC, pp. 1-8 (2016)
15. Canizes, B., Silva, M., Faria, P., Ramos, S., Vale, Z.: Resource scheduling in residential microgrids considering energy selling to external players. In: 2015 Clemson University Power Systems Conference (PSC), Clemson, SC, pp. 1-7 (2015)
16. Gomes, L., Abrishambaf, O., Faria, P., Vale, Z.: Retrofitting Approach for an Automated Load Testbed. In: ELECON Workshop – Dissemination & Transfer of knowledge (2016)

A Syntactic Treatment for Free and Bound Pronouns in a Type-Logical Grammar

María Inés Corbalán

Universidade Estadual de Campinas, São Paulo, Brazil
inescorbalan@yahoo.com.ar

Abstract. In this paper we modify the rule of proof for the pronominal connective of the **LLC** type-logical calculus [3] to distinguish between the free and the bound uses of personal pronouns in English.

Keywords: Pronouns · Lifted Pronominal Types · Substructural Logic · Jaeger's **LLC** calculus

1 Introduction

Pronouns, such as *she*, *her*, *herself*, are linguistic expressions that pick out their semantic value from other linguistic denotative expressions: the binder. Thus, the denotation of the binder is reused for the pronoun to obtain its reference. From a generative perspective, the licensing of pronominal expressions is determined by the so-called Principles A and B of the Binding Theory [1]. Principle A stipulates that an anaphora must be bound in its governing category (roughly, it must have a c-commanding local antecedent). Principle B stipulates that a pronoun must be free (i.e. not bound) within its governing category; this notwithstanding, a pronoun can be bound from outside this syntactic domain.[1] Thus personal pronouns also admit a free reading, in contrast with reflexive ones.

Categorial Grammars are formal grammars based on the Lambek **L** Calculus [4]. **L** is a substructural logic that rejects all of the structural rules of the Classic Sequent Calculus **SK**: Weakening (and also Expansion), Contraction and Exchange. Because of the lack of the Weakening and Contraction rules, Categorial Grammars are resource-conscious grammars, and the linguistic analysis they provide are resource-conscious proofs: all formulae occurring in categorial proofs have to be used at least once, and they cannot be reused. Consequently, bound pronouns pose a challenge for resource-conscious grammars. Several categorial —combinatory and type-logical— calculi have been proposed to deal with anaphora and anaphoric pronouns. Working on a Combinatory Categorial Grammar, Szabolcsi [6] assigns the doubling combinator W to reflexive pronouns. While Szabolcsi treats multiple-binding triggered by reflexives in the lexicon, Jaeger [3] uses a syntactic approach. Working on a Type-Logical Grammar, Jaeger develops the **LLC** calculus to process reflexive and personal pronouns in

[1] Roughly, Principle C stipulates that a referential expression must be free.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_27

a uniform way; he does not discriminate either syntactically (or semantically) among them. In other words, he does not take the Principles A and B of the Binding Theory into account.

In this paper, we modify **LLC** to give a more accurate treatment for personal pronouns. In a nutshell, we adopt the pronominal connective | from Jaeger, albeit changing its logical rules. We are interested in distinguishing, on the one hand, the free from the bound *uses* of a personal pronoun, and on the other, the bound (use of a) pronoun from the reflexive pronoun. As a consequence, although we deal with both reflexive and personal pronouns, our proposal is not intended to be a uniform approach.

The structure of the paper is as follows. In Section 2, we present the pronominal rules of the **LLC** calculus (in a sequent format) and we briefly discuss some questions related to the problem of overgeneration. In Section 3, we present our proposal and show how it can block some ungrammatical readings. We also expose the problem of Cut elimination posed by our pronominal rules. Section 4 concludes the paper.

2 LLC Calculus

The Lambek calculus with Limited Contraction (**LLC**) proposed by Jaeger [3] is a conservative extension of the Lambek-style core of Type-Logical Grammar. In a nutshell, **LLC** extends **L** by adding a third kind of slash type-constructor |, which compiles a limited version of the structural rule of Contraction into its left and right logical rules. Like Lambek calculus, **LLC** is free of structural rules.[2] The rules for | are displayed in Fig. 1.[3]

$$\frac{Y \Rightarrow M : A \qquad X, x : A, Z, y : B, W \Rightarrow N : C}{X, Y, Z, z : B|A, W \Rightarrow N[M/x][(zM)/y] : C} \; |L$$

$$\frac{X, x_1 : B_1, Y_1, \ldots, x_n : B_n, Y_n \Rightarrow N : C}{X, y_1 : B_1|A, Y_1, \ldots, y_n : B_n|A, Y_n \Rightarrow \lambda z.M[(y_1 z)/x_1] \ldots [(y_n z)/x_n] : C|A} \; |R$$

Fig. 1. Left and Right rules for |

When A in |L is a basic (also atomic) type, the left premise is an instance of the axiom, and thus the rule can be simplified, as shown in Fig. 2.

[2] Both systems also admit the Cut rule, i.e. adding the Cut rule does not give rise to any new theorem.

[3] In this paper we are principally concerned with the syntax of the (third person) pronouns. All the same, it is important to recall that the λ-term (MN) is the result of applying the functor M to the argument N, and $N[M/x]$ is the result of replacing all free occurrences of x in N by the λ-term M.

$$\frac{X, x : A, Z, y : B, W \Rightarrow M : C}{X, x : A, Z, z : B|A, W \Rightarrow M[zx/y] : C} \,|\mathrm{L}$$

Fig. 2. Simplified Left rule for |

In **LLC** anaphoric expressions are assigned the type $B|A$; it works as a type B in the presence of an antecedent of type A. In particular, personal and reflexive pronouns are uniformly assigned the type $n|n$.[4] In semantic terms, a pronoun denotes the identity function $\lambda x.x$ over individuals; the reference of a pronoun is identical with the reference of its nominal antecedent.

Since the pronominal type $n|n$ can be constructed by using either the $|R$ or the (simplified) $|L$ rule, and since reflexive and personal pronouns are assigned the same syntactic type (and the same semantic category), the system can accurately recognize the double reading for the pronoun *he* in a sentence like (1). Even so, **LLC** also triggers some oddities: a sentence like (2) may have the same derivation than, and can be read as, (3); the pair (4)-(5), as well as the pair (2)-(4), may present the same syntactic type $s|n$; and so forth.

1. John_1 said $\text{he}_{1/2}$ runs.
2. John likes him.
3. John likes himself.
4. He likes him.
5. *Him likes he.

3 Proposal: LLCE Calculus

We adopt the pronominal connective | from **LLC** to categorize pronominal expressions. Departing from Jaeger, and following previous ideas from Lambek [4], we assign lifted syntactic types derived from that of Jaeger —$n|n$— for personal pronouns. In effect, in **LLC** the following sequents can be derived:

6. $n|n \Rightarrow (s/n) \setminus (s|n)$
7. $n|n \Rightarrow (s|n) / (n\setminus s)$

From the semantic perspective, we follow Jaeger in adopting the identity function for personal pronouns, while we follow Szabolcsi [6] for the case of reflexive pronouns. We agree with her in that a reflexive pronoun is an operation that identifies the argument of the function it applies to.[5] In sum, we assume the following lexical entries for pronouns:

she/he: $(s|n)/(n\setminus s)$: $\lambda F.F$
her/him: $(s/n)\setminus(s|n)$: $\lambda F.F$

[4] As usual, we use n for nominals, and s for sentences.
[5] Alternatively, we may use the discontinuous type $((n\setminus s) \uparrow\ n) \downarrow (n\setminus s)$ with the same semantic for non-peripheral reflexives (cf. [5]).

herself/himself: $((n\backslash s)/n)\backslash(n\backslash s)$: $\lambda F.\lambda x.((Fx)x)$

Both subject and object pronouns return an unsaturated type $s|n$, that is, a sentence with a pronominal gap; the difference between them depends on the complex constituent they consume: an intransitive verb phrase —$n\backslash s$— or the non-standard constituent subject+verb —s/n. Since reflexives are assigned lifted Lambek-types, we do not need additional rules to process them; we only need the rules of **L**. In fact, in Szabolcsi's proposal, obligatory binding is not a consequence of a syntactic operation, but is triggered by the semantics of the reflexive pronoun. For pronominal types we propose the |L rule of [3] (which we rename $|\mathrm{L}_1$) and the rules in Fig. 3, which modify |R of **LLC**. We call our version of the Jaeger's system **LLCE**: Lambek calculus with Limited Contraction and Expansion.

$$\frac{X, x : A, Z, y : B, W \Rightarrow M : C}{X, x : A, Z, z : B|A, W \Rightarrow M[zx/y] : C}\,|L_1 \qquad \frac{X, Z, y : B, W \Rightarrow M : C}{X, x :¡A, Z, z : B|A, W \Rightarrow M[zx/y] : C}\,|L_2$$

$$\frac{X, x :¡A, Y \Rightarrow N : C}{X, Y \Rightarrow \lambda x.N : C|A}\,|R$$

Fig. 3. Left and Right rules for pronouns

Thus, **LLCE** contains two left rules for the pronominal type constructor |. The $|\mathrm{L}_1$ rule is Jaeger's own rule; it expresses the fact that for an anaphoric expression to be bound it needs an antecedent within the same structure, that is, in a *local* syntactic domain. This rule compiles a restricted version of the structural rule of (long-distance) Contraction, in that the antecedent A for the anaphora $B|A$ is doubly-consumed.[6] In addition, we split the |R rule of **LLC** obtaining a second left rule. The $|\mathrm{L}_2$ rule expresses that a free pronoun $B|A$ can be used if it is assumed an antecedent A not occurring in the local syntactic domain of the pronoun.[7]. This rule encodes the structural rule of Expansion, in that the antecedent for the pronoun is not consumed. The |R rule exhibits the fact that the occurrence of a free pronoun turns the complex constituent in which it occurs into a syntactically unsaturated one. Like in Jaeger's proposal, here too the rule of proof for a free pronominal type goes hand in hand with its free use in **LLCE**.

In fact, note that by using $|L_2$ and $|R$ of **LLCE**, we can derive the $|R$ rule of **LLC** for the particular case where $n = 1$. As an example, compare the (labeled) proofs for the sequent $s|n \Rightarrow s|n$ in **LLC** and **LLCE** (see Fig. 4).

[6] In other terms, this form of Contraction subsumes the structural rule of Permutation.
[7] This rule resembles the BIR rule of [2]. Hepple's system is not Cut-free.

$$\frac{x_1 : s \Rightarrow x_1 : s}{y : s|n \Rightarrow \lambda z.x_1[(yz)/x_1] : s|n}\,|R$$

$$\frac{\dfrac{x_1 : s \Rightarrow x_1 : s}{z : ¡n, y : s|n \Rightarrow x_1[(yz)/x_1] : s}\,|L_2}{y : s|n \Rightarrow \lambda z.x_1[(yz)/x_1] : s|n}\,|R$$

Fig. 4. Derivation for $s|n \Rightarrow s|n$ in **LLC** and **LLCE**

However, the |R rule of **LLCE**, unlike that of **LLC**, does not simultaneously construct a pronominal type to the left and to the right sides of the sequent. In this respect, our |R rule is syntactically and semantically analogous to the right rules for the standard slash connectives.

By breaking down the |R rule of **LLC** we exhibit more clearly that *free* and *bound* are labels that arise from the procedures by which we use a pronoun: we apply the $|L_1$ rule to get a bound use of a pronoun, while in applying the $|L_2$ we use a pronoun freely. The bound or the free reading for a personal pronoun is, basically, a consequence of a syntactic operation. The fact that a complex constituent became an unsaturated syntactic category is only a result of the occurrence of a free pronoun within the former.

Since we assign different types for reflexives and pronouns, and also for subject and object pronouns, **LLCE** adequately recognizes the grammatical sentences and blocks the ungrammatical ones below. As an example, consider the proofs in Figs. 5–8.[8]

8. John$_1$ said he$_{1/2}$ runs.
9. John$_1$ said Mary likes him$_{1/2}$
10. John$_1$ likes him$_{*1/2}$.
11. John$_1$ likes himself$_{1/*2}$.
12. *He$_1$ likes him$_1$/himself$_1$.

[8] We remind the reader that $\lambda x.Mx = M$, if x does not appear free in M (i.e. η-conversion). Thus, in Fig. 7, we replace $\lambda y_1.y_2y_1$ by y_2.

$$\dfrac{\begin{array}{cc}\vdots & \dfrac{\begin{array}{c}\vdots\\ x_1 : n, z : (n\backslash s)/s, x_2 : s \Rightarrow (zx_2)x_1 : s\end{array}}{x_1 : n, z : (n\backslash s)/s, y : \boxed{s|n} \Rightarrow (zx_2)x_1[(yx_1)/x_2] : s}\,|\mathrm{L}_1 \\ y_3 : n\backslash s \Rightarrow y_3 : n\backslash s & \end{array}}{x_1 : n, z : (n\backslash s)/s, w : \boxed{(s|n)/(n\backslash s)}, y_3 : n\backslash s \Rightarrow (z(yx_1))x_1[(wy_3)/y] : s}\,/L$$

Fig. 5. Derivation for (8): *John*$_1$ *said he*$_1$ *runs*: $((said'(run'j))j)$

$$\dfrac{\begin{array}{cc}\vdots & \dfrac{\begin{array}{c}\vdots\\ x_1 : n, z : (n\backslash s)/s, x_2 : s \Rightarrow (zx_2)x_1 : s\end{array}}{\dfrac{y_2 : ¡n, x_1 : n, z : (n\backslash s)/s, y_1 : \boxed{s|n} \Rightarrow (zx_2)x_1[(y_1y_2)/x_2] : s}{x_1 : n, z : (n\backslash s)/s, y_1 : s|n \Rightarrow \lambda y_2.(z(y_1y_2))x_1 : \boxed{s|n}}\,|R}\,|\mathrm{L}_2 \\ y_3 : n\backslash s \Rightarrow y_3 : n\backslash s & \end{array}}{x_1 : n, z : (n\backslash s)/s, w : \boxed{(s|n)/(n\backslash s)}, y_3 : n\backslash s \Rightarrow \lambda y_2.(z(y_1y_2))x_1[(wy_3)/y_1] : s|n}\,/L$$

Fig. 6. Derivation for (8): *John*$_1$ *said he*$_2$ *runs*: $\lambda y_2.((said'(run'y_2))j)$

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ x_1 : n, z : (n\backslash s)/n, x_2 : n \Rightarrow (zx_2)x_1 : s\end{array}}{x_1 : n, z : (n\backslash s)/n \Rightarrow \lambda x_2.(zx_2)x_1 : \boxed{s/n}}\,/R \qquad \dfrac{\dfrac{x_3 : s \Rightarrow x_3 : s}{y_1 : ¡n, y_2 : \boxed{s|n} \Rightarrow x_3[(y_2y_1)/x_3] : s}\,|\mathrm{L}_2}{y_2 : s|n \Rightarrow \lambda y_1.y_2y_1 : \boxed{s|n}}\,|R}{x_1 : n, z : (n\backslash s)/n, w : \boxed{(s/n)\backslash(s|n)} \Rightarrow y_2[(w\lambda x_2.(zx_2)x_1)/y_2] : s|n}\,\backslash L$$

Fig. 7. Derivation for (10): *John*$_1$ *likes him*$_2$*:* $\lambda x_2.((like'x_2)j)$

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ n, (n\backslash s)/n, n \Rightarrow s\end{array}}{n, (n\backslash s)/n \Rightarrow \boxed{s/n}}\,/R \qquad \begin{array}{c}*\\ s|n \Rightarrow s\end{array}}{n, (n\backslash s)/n, \boxed{(s/n)\backslash(s|n)} \Rightarrow s}\,\backslash L$$

Fig. 8. Illicit derivation for (10): *John*$_1$ *likes him*$_1$

3.1 The Problem of Cut Elimination

The proof for the Cut elimination theorem seems to require the use of the following restricted versions of the Permutation and Contraction rules:

$$\frac{X, A, ¡A, Z \Rightarrow C}{X, A, Z \Rightarrow C}\, C \qquad\qquad \frac{X, Y, ¡A, Z \Rightarrow C}{X, ¡A, Y, Z \Rightarrow C}\, P$$

Fig. 9. Structural rules for ¡

In order to prove Cut Elimination for **LLCE** we have to consider three more cases for principal Cut: the left premise of Cut is the conclusion of $|L_1$, of $|L_2$ or the conclusion of the Contraction rule for ¡, and the right premise is a conclusion of $|R$. The two former configurations are schematically given in Figs. 10 and 11. In either case, the principal Cut is replaced by a Cut of lower degree. The last configuration is more problematic, since the Contraction rule for ¡ introduces an extra (marked) occurrence of A, as showed in Fig. 12.

$$\dfrac{\dfrac{X_1, ¡A, Z_1 \Rightarrow C}{X_1, Z_1 \Rightarrow C|A}\,|R \qquad \dfrac{X_2, A, Z_2, C, W_2 \Rightarrow D}{X_2, A, Z_2, C|A, W_2 \Rightarrow D}\,|L_1}{X_2, A, Z_2, X_1, Z_1, W_2 \Rightarrow D}\,Cut$$

$$\rightsquigarrow$$

$$\dfrac{X_1, ¡A, Z_1 \Rightarrow C \qquad X_2, A, Z_2, C, W_2 \Rightarrow D}{\dfrac{\dfrac{X_2, A, Z_2, X_1, ¡A, Z_1, W_2 \Rightarrow D}{X_2, A, ¡A, Z_2, X_1, Z_1, W_2 \Rightarrow D}\,¡P}{X_2, A, Z_2, X_1, Z_1, W_2 \Rightarrow D}\,¡C}\,Cut$$

Fig. 10. Principal Cut for |: $|L_1$

$$\dfrac{\dfrac{X_1, ¡A, Z_1 \Rightarrow C}{X_1, Z_1 \Rightarrow C|A}\,|R \qquad \dfrac{X_2, Z_2, C, W_2 \Rightarrow D}{X_2, ¡A, Z_2, C|A, W_2 \Rightarrow D}\,|L_2}{X_2, ¡A, Z_2, X_1, Z_1, W_2 \Rightarrow D}\,Cut$$

$$\rightsquigarrow$$

$$\dfrac{X_1, ¡A, Z_1 \Rightarrow C \qquad X_2, Z_2, C, W_2 \Rightarrow D}{\dfrac{X_2, Z_2, X_1, ¡A, Z_1, W_2 \Rightarrow D}{X_2, ¡A, Z_2, X_1, Z_1, W_2 \Rightarrow D}\,¡P}\,Cut$$

Fig. 11. Principal Cut for |: $|L_2$

$$\dfrac{X_1, Z_1 \Rightarrow A \qquad \dfrac{X_2, A, ¡A, Z_2 \Rightarrow D}{X_2, A, Z_2 \Rightarrow D}\, C}{X_2, X_1, Z_1, Z_2 \Rightarrow D}\, Cut$$

Fig. 12. Principal Cut for |: $C_¡$

4 Conclusions

From an empirical point of view, **LLCE** seems to be more adequate than **LLC** to specifically deal with pronominal expressions in English. Indeed, in **LLCE** we are able to discriminate the anaphoric use of a personal pronoun from its free use. In addition, the pronominal rules of **LLCE** display a record of the quantity of free pronominal gaps a sentence contains. Nevertheless, unlike Jaeger's system, in **LLCE** the Cut rule does not seem to be redundant (i.e. admissible). Given that proving the Cut Elimination theorem requires adding (controlled version of) structural rules, it seems we have two options: either we assume that **LLCE** does not contain the Cut rule at all or we adopt a version of the Cut rule that encodes the Contraction rule and thus simultaneously deletes the occurrences of A and $¡A$.[9] The problem of proving Cut Elimination by using a this non-standard Cut rule is left open for future research. However, since the pronominal rules hold (a version of) the subformula property, we still have hope of succeeding in a bottom-up search for the proofs of **LLCE**.

References

1. Noam Chomsky. *Lectures on Government and Binding*. Kluwer, Dordrecht, 1981.
2. Mark Hepple. The grammar and processing of order and dependency: A categorial approach. *Phd, University of Edinburgh*, 1990.
3. Gehard Jäeger. *Anaphora and Type Logical Grammar*. Springer, 2005.
4. Joachim Lambek. The mathematics of sentence structure. *The American Mathematical Monthly*, 65(3):154–170, 1958.
5. Glyn Morrill and Oriol Valentín. Displacement logic for anaphora. *Journal of Computer and System Sciences*, 80(2):390–409, 2014.
6. Anna Szabolcsi. Bound variables in syntax (are there any?). 1987.

Acknowledgment: The research reported in the present paper was supported by a Doctoral scholarship granted by FAPESP (Process number 2013/08115-1). The author has profited from discussion with their supervisor Prof. Marcelo E. Coniglio (CNPq, Brazil, research grant number 308524/2014-4), and would like to thank the three anonymous DCAI'17 reviewers for comments and suggestions. All errors are mine.

[9] Note that any rule introduces an occurrence of $¡A$ in the right side of a sequent.

Personal Peculiarity Classification of Flat Finishing Skill Training by using Torus type Self-Organizing Maps

Masaru Teranishi, Shinpei Matsumoto, Nobuto Fujimoto and Hidetoshi Takeno

Hiroshima Institute of Technology
teranisi@cc.it-hiroshima.ac.jp

Abstract. The paper proposes an unsupervised classification method for peculiarities of flat finishing motion with an iron file, measured by a 3D stylus. The classified personal peculiarities are used to correct learner's finishing motions effectively for skill training. In the case of such skill training, the number of classes of peculiarity is unknown. A torus type Self-Organizing Maps is effectively used to classify such unknown number of classes of peculiarity patterns.
Experimental results of the classification with measured data of an expert and sixteen learners show effectiveness of the proposed method.

1 Introduction

In the technical education of junior high schools in Japan, new educational tools and materials are in development, for the purpose to transfer kinds of crafting technology. When a learner studies technical skills by using the educational tools, two practices are considered to be important: (1) to imitate motions of experts and (2) to notice their own "Peculiarity", and correct it with appropriate aids.

However, present educational materials are not yet effective to assist the practices because most materials consist still or motion pictures of tool motions of experts. Even though the learners could read out rough outlines of the correct motion from these materials, it is difficult to imitate detailed motion due to less information these materials have. Especially, it is most difficult to imitate fine motions and postures of the tool by only looking the expert motions from a fixed viewpoint. Furthermore, the learners could not recognize their own "peculiarity" as a difference between the expert's motion and the learner's motion from the materials. To solve the problem, a new assistant system for a brush coating skill has developed[1]. The system presents a learner corrective suggestion by play-backing the learner's motion by using animated 3D Graphics.

The paper describes the development of a new technical educational assistant system[2–4] that let learners acquire a flat finishing skill with an iron file. The system measures a flat finishing motion of a learner by a 3-D stylus device, and classifies the learner's "peculiarity". The system assists the learners how to correct bad peculiarities based on detected "peculiarity" classified from difference of the motions between the expert and the learner. The paper mainly

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_28

describes the learner's peculiarities classification method which is implemented in the proposed system. A torus type Self-Organizing Maps(SOM)[5, 6] is proposed to classify learners' "peculiarity" effectively.

2 Motion measuring system for flat finishing skill training

Fig.1(a) shows an outlook of the motion measuring devices of the system for flat finishing skill training. The system simulates a flat finishing task that flatten a top surface of an pillar object by an iron file. The system measures a 3D + time motion data of the file by using a 3D stylus. We use the PHANTOM Omni (SensAble Technologies) haptics device as the 3D stylus component of the system. The file motion is measured by attaching the grip of the file to the encoder stylus part of the haptics device. Assuming that the system will have a force feed back teaching function, we use a light weight mock file made of an acrylic plate which imitates a real 200mm length iron file[2, 3].

In the measurement task, a learner operates the mock file in order to flatten the top surface of a dummy work whose area has 25mm width and 25mm depth, at 80mm height. The system measures the motion of the mock file. The measured motion is recorded as a time series of the position of the file with X, Y, and Z coordinate values, and the posture with the tilt angles along the three axes with Tx, Ty, Tz as the radians. The spatial axes of the operational space and tilt angles of the file are assigned as shown in Fig.1(b).

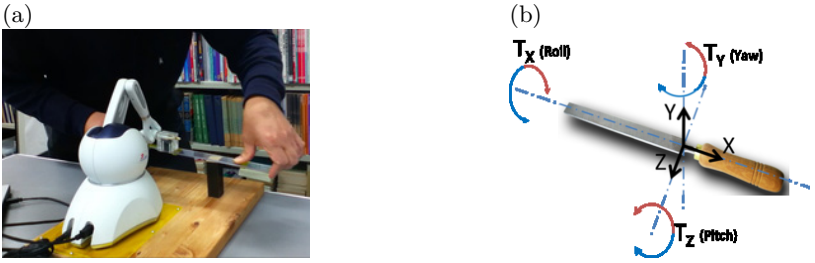


Fig. 1. Flat finishing skill measuring system: (a) outlook, (b) coordinates of measuring.

Fig.2 shows an example of an expert's motion measured by the system. In these plots, the expert operates the file with reciprocating motion 4 times within 20 seconds. The main direction of the reciprocating motion is along X axis. The expert pushes the file to the direction that X reduces, and pull the file to the opposite direction. Since the file works only in the pushing motion, we focused the pushing motion in the classification task in the rest part of the paper.

To classify every file-pushing motion as the same dimensional vector by the SOM, we have to preprocess measured data by two steps: (1) clipping out each time series potion of file-pushing motions, and (2) re-sampling the time series

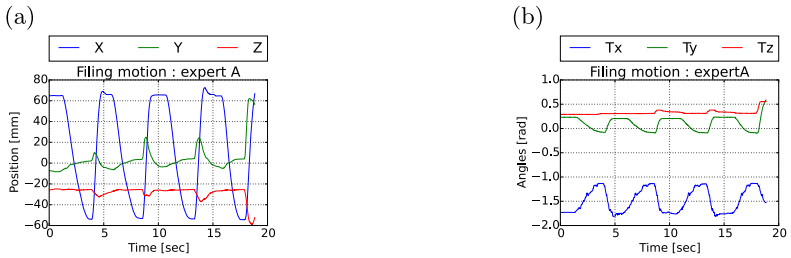


Fig. 2. Filing motion of an expert: (a) tool position, (b) posture of the tool.

portions. **(1) Clipping:** The clipping is done according to the file-pushing motion range which is defined in X coordinates, beginning with X_{begin}, and ending with X_{end}. **(2) Re-sampling:** The resulted clipped time series of the pushing motions have different time lengths. So we could not use the series directly to the SOM because the dimensions of each series differ. Therefore, we should arrange dimensions of all series to the same number. Every series are re-sampled in order to have the same number of the sampling time points, by using the linear interpolation, as shown in Fig.3(a). Since the stylus samples the motion enough fast, we can use the linear interpolation in the re-sampling with less lose of location precision. Fig.3(b) shows the re-sampled result of the expert motion, Fig.3(c) is that of a learner. In each plot, all four motions are imposed.

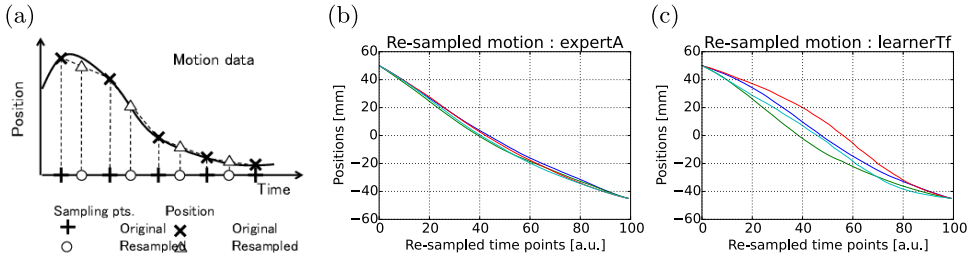


Fig. 3. (a) Re-sampling of a motion pattern, and re-sampled motion:(b) an expert, (c) a learner.

There is relevant differences between the expert's motions and the learner's motions. Every motion of the expert draws almost the same shaped curve, but the learner's don't. Each curves of learner's motion differs each time, in spite of identical person's actions. Additionally, the shape of curve of the expert means that the file is not moved in constant velocity. The expert operates the file by varying its velocity slightly. We call the slight varying velocity "velocity time series (VTS)", and consider it as the important factor of the expert motion.

The difference described above seems to be the main key of the correction in the skill teaching tasks. So we address a classification study of this difference

as the starting point of the proposed system. In this paper, we use only the X coordinate values of the measured data for the classification to keep simplicity of data computation. The aim of the classification study is to form the SOM to organize such different motion curves in the map. After that, taking and using code-books of the learner's motion as the "bad peculiarities".

3 Feature extraction of filing motion based on velocity time series

Velocity time series (VTS):
Although the filing motion obtained the former section have useful information about peculiarities, it is difficult to tell the learner their peculiarities exactly by only using the motion curve. Instead, to display the peculiarities on the basis of velocity time series (VTS) is an effective way. In this paper, we obtain a VTS of a re-sampled motion time series by computing local velocities at every re-sampled time point, by using 1st order differential approximation. Fig.4(a) shows VTSs of an expert, and Fig.4(b) shows VTSs of a learner. Since a VTS represents how velocities were taken at each time point of a filing motion, then learners could recognize and replay their motions more easily than looking raw motion plots. Additionally, learners also can imitate the expert's "model" motion. The average vector of an expert's VTSs is used as the "model" VTS.

Difference of velocity time series (dVTS):
Although the "model" VTS of Fig.4(a) is the model motion learners should imitate, a better way is to take difference of VTSs between the model and the learner for a learner to recognize and correct the peculiarities. For the purpose, we use difference of VTS (dVTS) as the feature pattern of the peculiarity. The dVTS is calculated by subtracting "model" VTS from the learners VTS.

Fig.4(c) shows dVTSs of the expert, Fig.4(d) shows that of the learner. The proposed system presents a learner dVTS. The dVTS tells the learner the peculiarity and the corrective points of the motion clearly. The dVTS displays a gap of VTSs between the model and the learner: if a learner imitates the model correctly, the dVTS draws a flat line, which means "Your skill is exact. No peculiarity", otherwise, for example, if the former part of the dVTS takes negative values, a trainer could point out the corrective point by telling the learner "You should move the file more fast in the former part of the motion".

4 Peculiarity classification by torus type SOM

We classify the file motion data by using the Self-Organizing Maps (SOM). Technical issues about the classification of the file motion are considered in three major points: (1)peculiarities are implicitly existing among the motion data, (2) the number of peculiarity variations, i.e., the number of classes is unknown and (3) every motions are not exactly same even though in the same learner, there are some fluctuation in each motion.

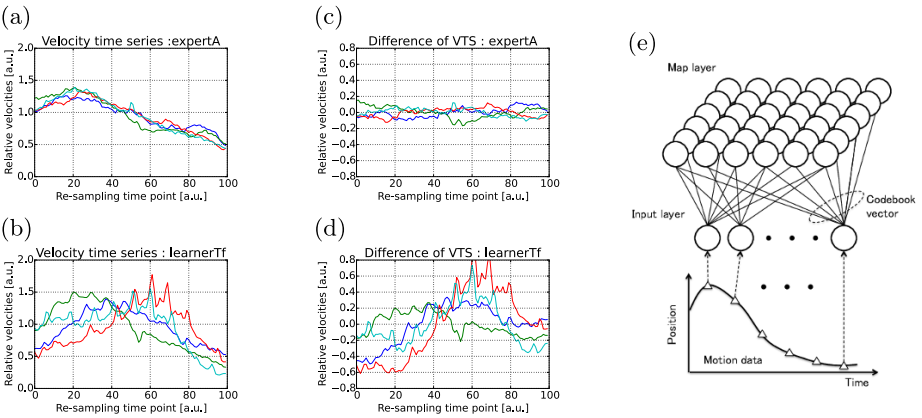


Fig. 4. Velocity time series (VTS) of (a) an expert, (b) a learner, difference of VTS (dVTS) of (c) the expert, (d) the learner, and (e) structure of SOM.

4.1 Structure of SOM

SOM is one of effective classification tools for patterns whose number of classes is unknown and whose classification features are implicit. Fig.4(e) shows the structure of the SOM. The SOM is a kind of neural networks which have two layers: one is the input layer, the other is the map layer. The input layer has n neuron units, where n is the dimension of input vector $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)^T$. The map layer consists of neuron units, which is arranged in 2D shape. Every unit of the map layer has full connection to all units of the input layer. The ith map unit u_i^{map} has full connection vector $\boldsymbol{m}_i$ which is called "code-book vector". The SOM classifies the input pattern by choosing the "firing code-book" $\boldsymbol{m}_c$ which is the nearest to the input vector $\boldsymbol{x}$ in the meaning of the distance defined as:

$$\|\boldsymbol{x} - \boldsymbol{m}_c\| = \min_i \{\|\boldsymbol{x} - \boldsymbol{m}_i\|\}. \tag{1}$$

The SOM classifies the high dimensional input pattern vector according to the similarity to the code-book vectors. The map units also arranged in two dimensional grid like shape, and neighbor units have similar code-book vectors. Therefore, the SOM is able to "map" and visualize the distribution of high dimensional input patterns into a simple two dimensional map easily. Since the SOM also could form a classification of patterns automatically by using "Self-Organizing" process described in the later section. Therefore, we need not to consider the number of classes.

The SOM organizes a map by executing the following three steps onto every input pattern: (1) first, the SOM input a pattern, (2) then it finds a "firing unit" by applying Eq.(1) to every code-book vector $\boldsymbol{m}_i$, (3) and it modifies code-book vectors of the "firing unit" and its neighbors. In the step (3), code-book vectors are modified toward the input pattern vector. The amount of modification is computed by the following equations, according to a "neighbor function" h_{ci}

which is defined based on a distance between each unit and the firing unit.

$$\boldsymbol{m}_i(t+1) = \boldsymbol{m}_i(t) + h_{ci}(t)\{\boldsymbol{x}(t) - \boldsymbol{m}_i(t)\} \tag{2}$$

where t is the current and $t+1$ is the next count of the modification iterations. The neighbor function h_{ci} is a function to limit modifications of code-book vectors to local map units which are neighbor the firing unit. The proposed method uses "Gaussian" type neighbor function. The Gaussian type modifies code-book vectors with varying amounts that decays like Gaussian function, proportional to the distance from the firing unit as follows:

$$h_{ci} = \alpha(t) \exp\left(-\frac{\|\boldsymbol{r}_c - \boldsymbol{r}_i\|}{2\sigma^2(t)}\right) \tag{3}$$

where $\alpha(t)$ is a relaxation coefficient of modification amount. The standard deviation of the Gaussian function is determined by $\sigma(t)$. We decrease both $\alpha(t)$ and $\sigma(t)$ monotonically as the modification iteration proceeds. The $\boldsymbol{r}_c, \boldsymbol{r}_i$ are the locations of the firing unit and the modified code-book vector unit, respectively. The reason why we use the Gaussian function is based on the assumption that the dVTSs distribute continuously in the feature space.

4.2 Torus type SOM

When we use the SOM straightforward, a "Map edge distortion" problem often occurs. The problem is observed as over-gathering of input data to code-books which are located edges of the map. Therefore, the classification performance become worth due to the each of the edge code-books represents more than one appropriate class of the input data. We have confirmed the problem in the early development of the system[3]. The problem is caused based on the fact that there is no code-book outside the edge, therefore the edge code-book can't be modified to appropriate direction in the feature space.

We solve the problem by introducing the torus type SOM[5, 6]. In the torus SOM, each code-book has cyclic neighbor relation in the feature map. In Fig. 4(e), every code-book at the edges of the map adjoins ones at the opposite edges. Since the torus SOM has no map edge, the map is free from edge distortion problem. Additionally, in case of 2-D visualization of the map, we can choose any code-book as the center of the map. Since the study aims to classify and visualize the distribution of correct motion and bad peculiarities, it is more understandable when the correct motion is located at the center. Therefore, the torus SOM makes the visualization of the pattern distribution more convenient.

5 Classification Experiment

To evaluate of the effectiveness of the proposed method, we carried out experiments of the filing motion peculiarities classification.

The motion data were measured with an expert and sixteen learners. Every person operated three of four filing motions. Totally we used 66 motion data for classification. Each motion data is clipped out as in range $X_{begin} = 50(mm), X_{end} = -45(mm)$, and re-sampled at $n = 100$ sampling points.

The torus SOM consists the input layer with 100 units and the map layer with 49 code-book vectors, which are arranged in 2D formation with 7 by 7, hexa-topology. The self-organizing process started with initial relaxation coefficient $\alpha = 0.01$, initial extent of the neighbor function $\sigma = 7$, and iterated 10,000 times with the Gaussian type neighbor function. We also made a non-torus type of SOM with the same experimental setups as the conventional method.

The dVTS of an expert and learners are classified as shown in Fig.5. Learners' peculiarities are classified rough three classes in both methods.

We can confirm two advantages of the proposed method:(1) Efficiently use of feature map, (2) Reduction of incorrect classification.

(1) In the proposed method, the expert's motions are well concentrated three code-books, center of the map. On the other hand, the convention SOM distributed the expert's motion to little far code-books, failed to gather. Additionally, the feature map of the conventional SOM has large "valley" between the classes in center part of the map, and the valley include some code-books. Such included code-books can not contribute to classification, and lose classification accuracy by causing over-gathering patterns to the edge code-book vectors. On the other hand, the torus SOM used most of all code-books as effective templates.

(2) The conventional SOM tended to gather patterns of different classes to a code-book vectors, which causes map distortion and misclassification. Such tendencies are observed at the upper left and the upper right edges as shown in Fig.6. The proposed method reduce the misclassification by arranging such patterns to some proper neighboring code-books as shown inf Fig.6.

From these results, the torus SOM works effectively to classify and visualize motion peculiarities of the learners.

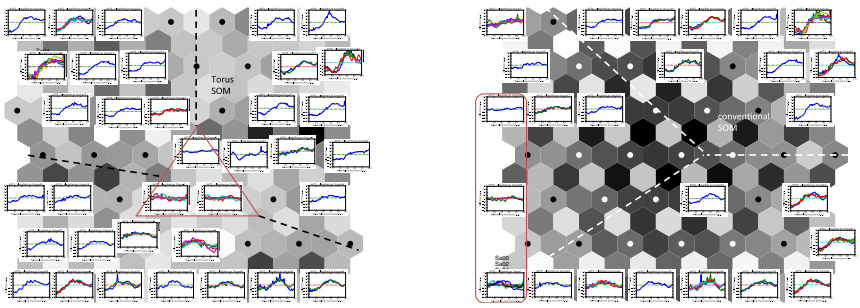


Fig. 5. Resulted feature map of torus SOM (left) and the conventional SOM(right). Plots surrounded with solid lines are the expert's motion. Dotted lines indicate rough boundary of classes. Each dot or plot are location of a code-book vector. The color of hexagons display distances between two code-book vectors: dark = long, light = short.

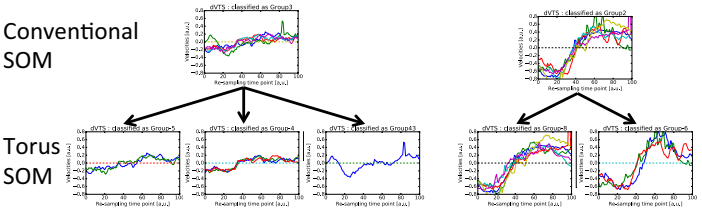


Fig. 6. Examples of the improved class: plots in upper row are incorrect classes of conventional SOM, plots in bottom row are corresponding classes in the torus SOM.

6 Conclusion

The paper proposed a new motion classification method of individual learner's "peculiarities" as a part of the development of the new technical educational tool of flat finishing skill. A torus Self-Organizing Maps(SOM) has proposed in order to classify the filing motion by using dVTS curves into an expert's motions and "peculiarities" of learners. The experimental results with measured data showed the effectiveness of the torus SOM to cope with "edge distortion" problem.

To evaluate further effectiveness of torus SOM, we have remained future works on data and learning viepoints. On the data viewpoints, the classification evaluation is required with more number of learner person data, and application of the method to other data components (Y, Z, Tx, Ty, Tz). On the learning viewpoints, we have to compare variant competitive learning algorithms which can be implemented on SOM.

References

1. Matsumoto, S., Fujimoto, N., Teranishi, M., Takeno, H., Tokuyasu, T.: A brush coating skill training system for manufacturing education at Japanese elementary and junior high schools. Artificial Life and Robotics, 21, 69–78 (2016)
2. Teranishi, M., Takeno, H., Matsumoto, S.: Classification of Personal Variation in Tool Motion for Flat Finishing Skill Training by Using Self-Organizing Maps. 2014 Ann. Conf. of Electronics, Information and Systems Society, I.E.E. of Japan, OS1'1-8, 1144–1147 (2014)
3. Teranishi, M., Matsumoto, S., Takeno, H.: Classification of Personal Variation in Tool Motion for Flat Finishing Skill Training by Using Self-Organizing Maps. The 16th SICE System Integration Division Annual Conference, 3L2-1, 2655–2669 (2015)
4. Matsumoto, S., Teranishi, M., Takeno, H.: A Training Support System of Brush Coating Skill with Haptic Device for Technical Education at Primary and Secondary School. Intl. Sym. on Art. Life and Robotics (AROB 20th 2015), GS10–5 (2015)
5. Kohonen, T.: Self-Organizing Maps. Springer (2001)
6. Ito, M., Miyoshi, T. and Masuyama, H.: The characteristics of the torus self-organizing map. 6th Intl. Conf. on Soft Computing (IIZUKA2000), pp.239–244 (2000)

A Block-separable Parallel Implementation for the Weighted Distribution Matching Similarity Measure

Mauricio Orozco-Alzate[1], Eduardo-José Villegas-Jaramillo[1], and Ana-Lorena Uribe-Hurtado[1⋆]

Universidad Nacional de Colombia - Sede Manizales, Facultad de Administración, Departamento de Informática y Computación, Grupo de Ambientes Inteligentes Adaptativos - GAIA, km 7 vía al Magdalena, Manizales 170003, Colombia
{morozcoa, ejvillegasj, alhurtadou}@unal.edu.co

Abstract. Automatic pattern recognition is often based on similarity measures between objects which are, sometimes, represented as high-dimensional feature vectors —for instance raw digital signals or high-resolution spectrograms. Depending on the application, when feature vectors turn extremely long, computations of the similarity measure might become impractical or even prohibitive. Fortunately, multi-core computer architectures are widely available nowadays and can be efficiently exploited to speed-up computations of similarity measures. In this paper, a block-separable version of the so-called *Weighted Distribution Matching* similarity measure is presented. This measure was recently proposed but has not been analyzed until now for a parallel implementation. Our analysis shows that this similarity measure can be easily decomposed into subproblems such that its parallel implementation provides a significant acceleration in comparison with its corresponding serial version. Both implementations are presented as Python programs for the sake of readability of the codes and reproducibility of the experiments.

Keywords: Similarity measure, parallel algorithm, weighted distribution matching, block separability, multi-core implementation

1 Introduction

Measuring the degree of similarity among objects is an essential issue in several pattern recognition fields such as information retrieval, clustering and classification. Objects in these fields are typically observed as digital signals (data sequences, images or videos) and afterwards represented as feature vectors which, according to their nature, can be compared by a wealth of similarity measures that has been proposed in the literature [2,8]. Such similarity measures must be fast, either by definition or implementation, as well as accurate to satisfactorily capture the often task-dependent concept of resemblance between objects.

⋆ Estudiante del Doctorado en Ingeniería, Industria y Organizaciones - Universidad Nacional de Colombia - Sede Manizales.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_29

The simplest and most straightforward feature representation consists in using the signal samples themselves, for instance the pixels in the image ordered as a long feature vector. A more elaborate but still simple representation is the computation of a histogram, which provides an estimation of the data distribution by partitioning the range of admissible sample values of the signal in bins and counting the number of samples that fall in each of them. A properly normalized version of the histogram can be considered an estimate of the probability mass/density function and, therefore, it is very useful to characterize and discriminate among observations.

Pixel representations and histograms tend to generate high-dimensional feature vectors. Consider, for instance, the following two exemplar cases: first, images from high-definition cameras such as the ones used in dermatopathology with typical resolutions of $920 \times 1080 = 993,600$ pixels and, second, full histograms for RGB color images whose three bands are encoded with one byte each and, therefore, $2^8 \times 2^8 \times 2^8 = 16,777,216$ different colors can be generated [3]. Even though high-dimensional feature representations exhibit known disadvantages such as non-convex subspaces and potential loss of information of the spatial connectivity, they are still valuable representations to be used, specially when sufficient training objects are available or in case that no good and more advanced features can be found [6].

Amongst the plethora of available similarity measures, the so-called *Weighted Distribution Matching* (WDM) [5] stands out due to its fulfillment of the abovementioned desired properties of speed and accuracy as well as because it takes into account the structure of the feature vector, namely the relationship between consecutive entries. In spite that the WDM similarity measure has a low computational cost, a detailed analysis of its mathematical structure —aimed at revealing possibilities for decomposing the algorithm into subproblems— has not been undertaken yet. Such an analysis will serve to formulate a parallel implementation of the WDM similarity measure that takes advantage of the nowadays omnipresent multi-core computer architectures with the aim of speeding up its computation when comparing high-dimensional feature vectors.

In this paper, we carry out an analysis of the mathematical formulation of the WDM similarity measure, particularly looking for block separability [4] such that partial results can be computed independently on different computer cores. For the sake of reproducibility and readability, Python implementations are presented. They, of course, could be speeded up further by translating the implementations to a compiled language if the user requires an optimized solution to be deployed instead of a flexible multi-core implementation for simulation purposes. The remaining part of the paper is organized as follows. The original sequential algorithm of the WDM similarity measure is presented in Section 2. The proposed block-separable version of it is described in Section 3. Experiments showing the performance of both sequential and parallel implementations of the WDM similarity measure, for comparing high-dimensional feature vectors as those resulting from high-resolution images and full color histograms, are presented in Section 4. Finally, our conclusions are drawn in Section 5.

2 Weighted Distribution Matching Similarity Measure

The WDM similarity measure was proposed in the recent paper by Correa-Morris et al. [5]. Given two feature vectors $\mathbf{x} = [x_0, \ldots, x_{N-1}]^\intercal$ and $\mathbf{y} = [y_0, \ldots, y_{N-1}]^\intercal$ to be compared, this measure is defined as follows:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{i=0}^{N-2} \{w(x_i, y_i)\delta([x_i, x_{i+1}]^\intercal, [y_i, y_{i+1}]^\intercal) - v(x_i, y_i)(1 - \delta([x_i, x_{i+1}]^\intercal, [y_i, y_{i+1}]^\intercal))\}, \tag{1}$$

where $w(x_i, y_i)$ is a reward weight and $v(x_i, y_i)$ is a penalty weight. Even though any pair of strictly increasing/decreasing monotonic functions can be used for these weights, the simplest and parameterless option [5] is defining the weights as follows:

$$w(x_i, y_i) = \frac{\min(x_i, y_i)}{\max(x_i, y_i)} \quad \text{and} \quad v(x_i, y_i) = 1 - w(x_i, y_i)$$

The function $\delta([x_i, x_{i+1}]^\intercal, [y_i, y_{i+1}]^\intercal)$ is given by

$$\delta([x_i, x_{i+1}]^\intercal, [y_i, y_{i+1}]^\intercal) = \begin{cases} 1, & \text{if } \operatorname{sign}(x_{i+1} - x_i) = \operatorname{sign}(y_{i+1} - y_i) \\ 0, & \text{otherwise.} \end{cases}, \tag{2}$$

where $\operatorname{sign}(x_{i+1} - x_i)$ and $\operatorname{sign}(y_{i+1} - y_i)$ are called the *local shapes* at i of $\mathbf{x}$ and $\mathbf{y}$, respectively. In words, $\delta([x_i, x_{i+1}]^\intercal, [y_i, y_{i+1}]^\intercal)$ either adds the reward weight if the local shapes at i are the same for both feature vectors or, alternatively, subtracts the penalty weight if they do not match.

It is convenient to introduce the following simplified notation in order to facilitate subsequent analyses carried out in Section 3: $\delta_i = \delta([x_i, x_{i+1}]^\intercal, [y_i, y_{i+1}]^\intercal)$, $w_i = w(x_i, y_i)$ and $v_i = v(x_i, y_i)$. Then, Equation (1) can be re-written as follows:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{i=0}^{N-2} \{w_i\delta_i - v_i(1 - \delta_i)\}. \tag{3}$$

3 A Block-separable Version of the Weighted Distribution Matching Similarity Measure

According to Censor and Zenios [4], the mathematical structure of an algorithm as well as the specific properties of the problem to which it is applied define its potential of being parallelizable or not. Regarding the problem structure, the first analysis to be performed is examining whether the problem is separable into P blocks, such that each block —a subproblem— can be computed in one of P processors. The block separability, for an arbitrary function $f(\mathbf{x})$, consists in the possibility of re-writing it as $f(\mathbf{x}) = \sum_{p=0}^{P-1} f_p\left(\mathbf{x}^{(p)}\right)$, where $\mathbf{x}^{(p)}$ is a subvector of $\mathbf{x}$, i.e. $\mathbf{x} = \left[\left(\mathbf{x}^{(0)}\right)^\intercal, \ldots, \left(\mathbf{x}^{(P-1)}\right)^\intercal\right]^\intercal$.

This block-separable structure can be easily generalized to several input variables; in that way and noticing from Equation (3) that the definition of the WDM similarity measure allows the use of the associative property of addition, $s(\mathbf{x}, \mathbf{y})$ can be written as follows:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{p=0}^{P-1} s\left(\mathbf{x}^{(p)}, \mathbf{y}^{(p)}\right), \tag{4}$$

where $\mathbf{x} = \left[\left(\mathbf{x}^{(0)}\right)^\intercal, \ldots, \left(\mathbf{x}^{(P-1)}\right)^\intercal\right]^\intercal$ and $\mathbf{y} = \left[\left(\mathbf{y}^{(0)}\right)^\intercal, \ldots, \left(\mathbf{y}^{(P-1)}\right)^\intercal\right]^\intercal$. Subvectors $\mathbf{x}^{(p)}$ and $\mathbf{y}^{(p)}$ $\forall p$ have the same length in the particular case that the length of the feature vectors (i.e. the number of features, denoted by N) is a multiple of the number of processors; therefore, subvectors can be accessed in a based-indexed mode such that Equations (3) and (4) can be combined as follows:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{p=0}^{P-1} \sum_{i=pm}^{(p+1)m-1} \{w_i \delta_i - v_i(1 - \delta_i)\}, \quad m = \frac{N}{P}. \tag{5}$$

However, a more general expression is required for the case when the length of the vectors is not a multiple of P. In that situation, a feature vector $\mathbf{x}$ can be partitioned into a collection of $P-1$ equal-length subvectors $\{\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(P-2)}\}$ and a last subvector $\mathbf{x}^{(P-1)}$ containing the remaining part of $\mathbf{x}$. The length of the latter is greater than or equal to the length of the former ones. Such a general expression is given by:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{p=0}^{P-2} \sum_{i=pm}^{(p+1)m-1} \{w_i \delta_i - v_i(1-\delta_i)\} + \sum_{i=(P-1)m}^{N-1} \{w_i \delta_i - v_i(1-\delta_i)\}, \quad m = \left\lfloor \frac{N}{P} \right\rfloor. \tag{6}$$

In words, the first term of Equation (6) accounts for the contributions of the first equal-length $P-1$ blocks while the last one corresponds to the last block.

4 Implementations and experiments

Both serial and parallel versions for computing the WDM similarity measure were implemented in Python. In order to make our experiments reproducible, we do not present pseudocodes but prefer to provide the actual Python programs; see Listings 1.1 and 1.2, respectively. The latter was built upon an example presented in [9]. In order to illustrate the advantage of the parallel version when comparing large feature vectors, we consider the length of full RGB color histograms ($2^8 \times 2^8 \times 2^8$) for the case study. Notice that, for simulation purposes, we generate two random vectors of the above-mentioned length whose values are filled in single code lines by using list comprehension [7]. Those random vectors, of course, might be easily replaced by pre-computed values corresponding to actual histograms (see lines 22 to 24 in Listing 1.2).

Listing 1.1. Serial implementation of the WDM similarity measure

```
import random, time, sys
random.seed(1)

def wdmSimilarity(x, y):
    wdm = 0
    for i in range(len(x)-1):
        localShapeX = x[i]-x[i+1]
        localShapeY = y[i]-y[i+1]
        w = min(x[i], y[i])/max(x[i], y[i])
        if (localShapeX > 0 and localShapeY < 0) or \
           (localShapeX < 0 and localShapeY > 0):
            wdm -= 1 - w
        else:
            wdm += w
    return wdm

if __name__ == '__main__':
    N = 2**24     # Bins in the color histogram of an 8-bit RGB color image
    x = [random.randrange(1,100,1) for _ in range (N)] # Sample random data
    y = [random.randrange(1,100,1) for _ in range (N)] # Sample random data

    start = time.time()                                # Start the chronometer
    result = wdmSimilarity(x, y)                           # Call the function
    elapsedTime = time.time()- start                   # Take the elapsed time

    print("The computation  took " + str(elapsedTime) + " seconds")
    print("The value of the wdm similarity is: " + str(result))
```

Listing 1.2. Parallel implementation of the WDM similarity measure

```
import random, time, sys
from multiprocessing import Pool, cpu_count
random.seed(1)

def wdmSimilarity(x, y):
    wdm = 0
    for i in range(len(x)-1):
        localShapeX = x[i]-x[i+1]
        localShapeY = y[i]-y[i+1]
        w = min(x[i], y[i])/max(x[i], y[i])
        if (localShapeX > 0 and localShapeY < 0) or \
           (localShapeX < 0 and localShapeY > 0):
            wdm -= 1 - w
        else:
            wdm += w
    return wdm

def blockTasks(subvectors):
    return wdmSimilarity(subvectors[0], subvectors[1])

if __name__ == '__main__':
    N = 2**24     # Bins in the color histogram of an 8-bit RGB color image
    x = [random.randrange(1,100,1) for _ in range (N)] # Sample random data
    y = [random.randrange(1,100,1) for _ in range (N)] # Sample random data

    P = cpu_count()        # Count the number of available processing cores
    start = time.time()                                # Start the chronometer
    pool = Pool(processes=P)       # Create a process Pool with P processes

    # Generation of indexes to split feature vectors into blocks:
    m = len(x)//P
    arg = [(x[p*m:(p+1)*m+1], y[p*m:(p+1)*m+1]) for p in range(P-1)]
    arg.append((x[(P-1)*m:], y[(P-1)*m:]))

    partialResults = pool.map(blockTasks, arg) # Map blockTasks to the Pool

```

```
37      finalResult = sum(partialResults)       # Perform the external summation
38
39      elapsedTime =  time.time() - start              # Take the elapsed time
40
41      print("The computation took " + str(elapsedTime) + " seconds")
42      print("The value of the wdm similarity is: " + str(finalResult))
```

The WDM similarity measure, as expressed in Equation 3 but for vectors having an arbitrary length, is defined in both codes as a function called `wdmSimilarity`. The serial implementation of the measure simply consists in the invocation of that function; see line 23 in Listing 1.1. In contrast, in order to implement the block-separable version that was presented in Equation 6, it is required to generate indexes defining the block partitions. Such indexes, along with the corresponding vector partitions, are generated in lines 32 and 33 in Listing 1.2. Note that, in those lines, we again profit from the efficiency of list comprehension. Besides, remember that list slices in Python are specified as left-closed, right-open intervals.

The vector partitions are stored in a list of arguments that are afterwards mapped to a pool of processors. We use methods from the `multiprocessing` module —as originally suggested in [9]— for counting the number of available processors and mapping the arguments to the corresponding processes; see lines 26, 28 and 35 in Listing 1.2. Finally, partial results returned by the parallel processes are summed up in order to produce the overall similarity value. Elapsed times are measured in both implementations by using the `time.time()` method.

Experiments were performed in two computer systems: a desktop with an AMD® A6-6310 quad-core processor and a server Intel® Xeon® with 32 CPUs. The first one is an example of a widely available personal computer; the second one, even though slower than the desktop, allows us to illustrate the parallelization with a larger number of processors. Results of the experiments performed in both systems are shown in Figures 1 and 2. We incrementally varied the number of processors, instead of just using all the available ones (as indicated by line 26 in Listing 1.2) in order to study the behavior as the number of processors increases.

The following observations can be made from the curves. Note in Figure 1 that, if the four cores of the Desktop are used, the processing time is reduced up to more than a half of the time required for a single processor. The enhancement observed on the server is even more evident, see Figure 2: from about 31 seconds for a single processor to a computing time that starts to reach a relatively stable value (around 5 seconds) at about 10 processors —this is roughly a six-fold acceleration. The largest enhancements on both machines are obtained when moving from one to two and three processors, respectively. Notice that the improvements shown in Fig. 2 are not linearly but exponentially decreasing. This is because parallelization follows the expected behavior governed by the Amdahl's law [1]. Moreover, since the computing cores share the same memory, they must to compete at some point for resources against other underlying processes, e.g. the unavoidable ones corresponding to the operative system. Several repetitions of the experiments did not reveal significant differences on the above-discussed behaviors.

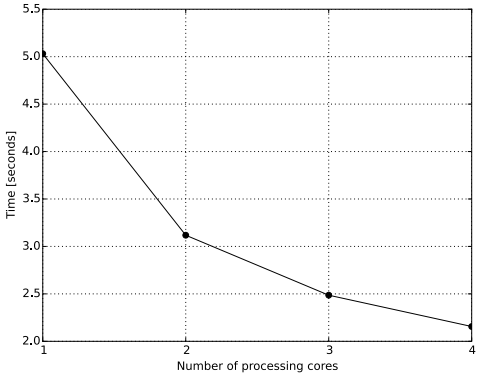


Fig. 1. Results obtained on a Desktop Hewlett Packard Pavilion 23, having an AMD® A6-6310 quad-core processor @ 1.80 GHz.

5 Conclusions

A block-separable version of the WDM similarity measure was proposed, aimed at its parallel implementation for the efficient similarity-based classification of extremely long feature vectors. This parallelization, according to our experimental evidence for vectors having lengths comparable to that of full RGB color histograms, allows to achieve speed-up factors of about 2 and 6 times when tested on typical multi-core personal computers and multi-processor servers, respectively. The simplicity of Python, along with methods from the `multiprocessing` module, makes it a very convenient tool to translate parallelized algorithms into simulation programs in a very straightforward way. Such programs, afterwards, can be of course accelerated further by rewriting them in a compiled language. As future work, we plan to explore the possibilities for parallelizing other recently proposed (di)similarity measures such as the Poisson-Binomial radius distance [10] and other state-of-the-art bin-to-bin and cross-bin distance measures.

Acknowledgments The authors acknowledge support to attend DCAI'17 provided by Universidad Nacional de Colombia through "Convocatoria para la Movilidad Internacional de la Universidad Nacional de Colombia 2016 - 2018". They also acknowledge project No. 32059 (Code Hermes) within "Convocatoria interna de investigación de la Facultad de Administración 2015".

References

1. Amdahl, G.M.: Validity of the single processor approach to achieving large scale computing capabilities, reprinted from the AFIPS Conference Proceedings, vol. 30 (Atlantic City, N.J., Apr. 1820), AFIPS press, Reston, va., 1967, pp. 483-485. IEEE Solid-State Circuits Society Newsletter 12(3), 19–20 (2007)

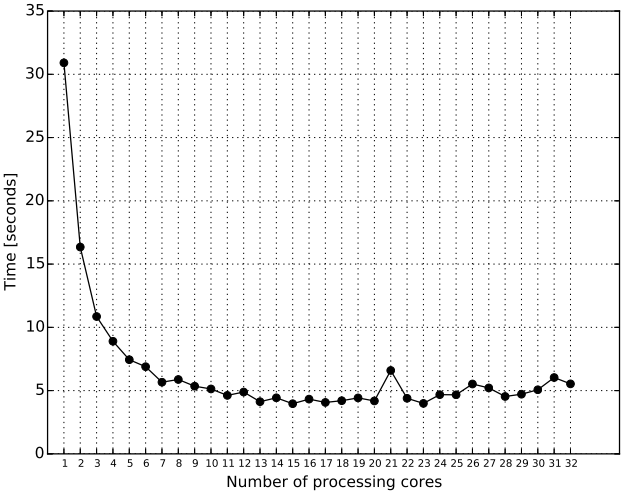


Fig. 2. Results obtained on a server Intel® Xeon® CPU E7- 4820 @ 2.00GHz, having 32 CPUs.

2. Bandyopadhyay, S., Saha, S.: Similarity Measures. In: Unsupervised Classification: Similarity Measures, Classical and Metaheuristic Approaches, and Applications, chap. 3, pp. 59–73. Springer, Berlin (2013)
3. Burger, W., Burge, M.J.: Color images. In: Principles of Digital Image Processing: Fundamental Techniques, pp. 1–47. Springer, London (2009)
4. Censor, Y., Zenios, S.A.: Parallel Optimization Methods. In: Handbook of Parallel Computing and Statistics, chap. 6, pp. 197–224. Statistics: A Series of Textbooks and Monographs, Chapman and Hall/CRC (2005)
5. Correa-Morris, J., Martínez-Díaz, Y., Hernández, N., Méndez-Vázquez, H.: Novel histograms kernels with structural properties. Pattern Recognition Letters 68, Part 1, 146 – 152 (2015)
6. Duin, R.P.W., Pekalska, E.: The dissimilarity representation for non-Euclidean pattern recognition, a tutorial. Delft University of Technology, The Netherlands. Available at `http://rduin.nl/presentations/DisRep_Tutorial.pdf` (2011)
7. Goldwasser, M.H., Letscher, D.: List comprehension. In: Object-Oriented Programming in Python, p. 148. Prentice Hall, New Jersey (2008)
8. Pekalska, E., Duin, R.P.W.: Dissimilarity measures. In: The Dissimilarity Representation for Pattern Recognition: Foundations and Applications, chap. 5, pp. 215–252. World Scientific, Singapore (2005)
9. Smith, R.W.: Multi-core and distributed programming in Python. Blog entry in Praetorian Security Blog. Available at `https://www.praetorian.com/blog/multi-core-and-distributed-programming-in-python` (aug 2012)
10. Swaminathan, M., Yadav, P.K., Piloto, O., Sjöblom, T., Cheong, I.: A new distance measure for non-identical data with application to image classification. Pattern Recognition 63, 384 – 396 (2017)

Artificial Curation for Creating Learners Manual based on Data Semantics and User Personality

Miki Ueno, Masataka Morishita, and Hitoshi Isahara

Toyohashi University of Technology
1-1 Hibarigaoka, Tempaku-cho, Toyohashi, Aichi, 441-8580, Japan
ueno@imc.tut.ac.jp

Abstract. Curation services contribute to create manual web pages for learners and teachers especially for e-Learning. However, it is difficult to determine the quality of web pages for certain learning purpose. In this paper, we propose the novel method of generating learners' manuals automatically based on semantic information of web pages and users personality. As an example, we implemented the application based on the proposed method and it was applied to the process of learning Git, which is one of popular version control systems. From our experiments, we discuss the relation between semantic of web pages and user personalities.

Keywords: Artificial curation, Learners manuals, User activity, Git, e-Learning, Pattern recognition

1 Introduction

Personal web pages are rapidly increasing, it causes the problem that it is difficult to find the valuable web pages. There are similar situations in various fields, especially educational fields. Currently, learners get information from textbook, lectures, and web pages. Some web service is specialized to *curation*, which is summarized several web pages about a certain topic[1] or certain question[2] by users. These web pages sometimes gives learners good suggestion as good manuals. Learners' manuals are significant for learners to obtain some new knowledges. However it is difficult for teachers to prepare and determine suitable manuals for all of their students. Some researchers have been reported about analyzing learners behavior[3] or generating manuals[4]. However, there are few research about generating manuals automatically base on learners behavior. To solve this problem, we propose a novel way of generating learners' manuals automatically according to semantics of web pages and learners personality. In this research, we focus on the process of leaning "Git[5]", which is one of popular version control systems, utilizing learners eye behavior from wearable devices[6] as an examples.

2 Basic Concept

Generally, students learn something based on their own way and get some information from their teachers. Nowadays, it has become easier to obtain lots of information for

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_30

learn something from the Internet. However, it is difficult for beginners to decide how to learn. Mostly, the effective way of learning depends on person to person. Learners manage to acquire their suitable way. In this research, we propose a method for automatic summary of information based on the frequency of referring to web pages during learning in order to support to acquire the way of learning for each person.

Some documents and images are extracted from existing web pages and are combined to reproduce manual. To decide importance of some document and images, following three modules are defined.

3 Proposed Method

The proposed method reproduce manuals from existing web pages which are called *candidate pages* in this paper. We define candidate pages set D_a for each learning topic a. Each candidate page $d_i^a \in D_a$. Candidate page set D is created from user activities.

3.1 Document Similarity Module

To avoid duplicate similar pages based on the semantic similarity between two documents by utilizing paragraph vector[7]. Each paragraph is extracted from each page based on tags of HTML. For two combination of candidate pages ${}_{|D_a|}\mathrm{C}_2$, the page similarities are calculated. If the page similarity is larger than the value of threshold, the value of page importance is calculated from image classification module described in 3.2, and which page is suitable for creating manual page.

3.2 Image Classification Module

This module determine the importance of each image e_j^d in each candidate page d_i^a. In this research, important images are strongly related to contents of pages; learning topic a. For examples, if candidate pages describe CUI commands, the concept images of command and examples images of command line in CUI console are important, but characters of commentators and ad banner are not so important. In this module, we define several classes C_a of images for each learning topic a, and collect example images for each class $c_k^a \in C_a$ from web pages. Then, convolutional neural network[8][9] are constructed to classify unknown input images.

3.3 User Activity Classification Module

We proposed the method to summarize web pages and generating learners manual based on the frequency of watching certain pages. In this paper, we focus on processes of learning Git which is the most modern version control system. While using git, users usually see explorers and git clients in the monitor. The process to generating manual pages is described as follows.

1. Prepare two monitors showing each browser in front of learners; User's input git commands into Learn Git Branching[10] on the left monitor and look up some information from the web pages on the right monitor. Learners wear a JINS MEME ES[6] glasses which has three types of sensors.

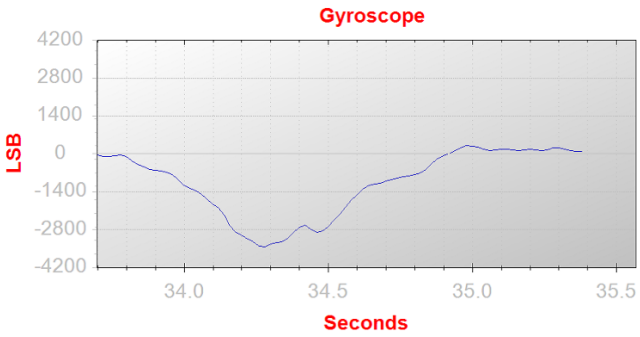


Fig. 1. The value of the Z axis when the learner's line of sight moves from the left to right window

2. When the eye tracking is moving from left to right and the value of the z-axis is over $z^{\text{threshold}}$[LSB] from gyroscope(gyro), get the URL of both of web pages and store the relationships between Learn Git Branching pages on the left monitor and referring pages on the right monitor.
3. Calculate the frequency of moving eye tracking for each set of left and right pages.
4. According to the above frequency, documents and image classifications described in above two modules, calculate the score of importance and get screen-shots of web pages which have scores higher than the threshold. Then, those images are listed into generating pages as a learner's manual. The order and size of each screen-shot are determined by the score.

In this method, the learner's manual are generated automatically only based on data semantics and learner's behavior. Figure 1, 2 shows the value of the Z axis when the learner's line of sight moves from the left to right window, the value of the Z axis when the learner's line of sight moves from the right to left window respectively. From preliminary experiment by two subjects(male, twenties), we set $z^{\text{threshold}} = \pm 2800$ in this paper.

4 Experiment

To define detailed states for user activity classification module, "$git rebase" command of version control system "Git[5]" is taken as an example in this experiment. "$git rebase" is a command for editing history of version control. Generally, "$git rebase" command is more difficult than other "Git" commands. We take the process of learning git on *Learn Git Branching*[10]. Learn Git Branching provides simulated a git client on the web pages. Without students installing a git client on their own computers, students use git commands on simulated git client only by using web browser. This experiments were carried out by three subjects(male, twenties); their knowledge about "$git rebase" are quite different.

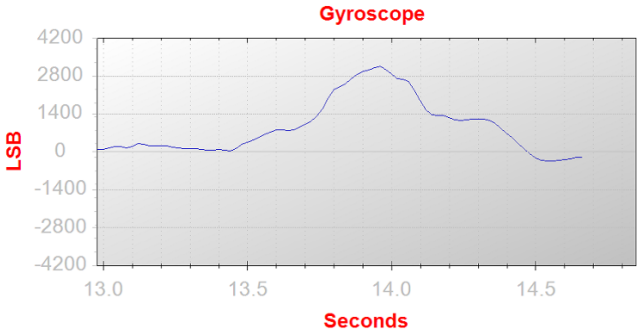


Fig. 2. The value of the Z axis when the learner's line of sight moves from the right to left window

We regarded three example problems of "various tips" in page "Select a level" as three kinds of learning topics. We did not set time limit. All of the example problems are related to "$git rebase" command.

Each user's knowledge about Git

User A
He is good at the concept of Git and "$git rebase". He use Git on CUI usually.

User B
He use Git on GUI usually, but he is not so good at the concept of Git. He does not remember basic git command on CUI.

User C
He understand basic concept of Git and some commands such as "$git add" and "git commit". However, he rarely use git and does not understand the concept of git branch.

5 Result and Discussion

From the experiment, kinds of pages which user referring to are divided as follows.

Articles Personal or corporate articles on various topic about "Git"
References Pages describes the concept of "Git" and systematized manipulations of commands.
Search Results pages of Google search engine.
QA pages on questions and answers about "Git"

Table 1 shows viewing time and URL of Top 5 reference pages by user. Table 2 shows types and viewing time of reference pages by user.

[1] "move" in Japanese
[2] "branch" and "move" in Japanese

Table 1. Viewing time and URL of Top 5 reference pages by user

Rank	User	URL	Viewing time[sec]
1	A	https://liginc.co.jp/web/tool/79390	81
	B	http://d.hatena.ne.jp/mrgoofy33/20100910/1284069468	412
	C	http://www.backlog.jp/git-guide/stepup/stepup2_3.html	135
2	A	http://www.backlog.jp/git-guide/stepup/stepup7_6.html	77
	B	https://www.google.co.jp/#q=git+master+move[1]	388
	C	http://www.backlog.jp/git-guide/stepup/stepup6_4.html	134
3	A	http://tkengo.github.io/blog/2013/05/16/git-rebase-reference/	43
	B	http://www.backlog.jp/git-guide/stepup/stepup2_5.html	347
	C	http://www.backlog.jp/git-guide/stepup/stepup2_2.html	134
4	A	https://www.google.co.jp/#q=git+cherry-pick	15
	B	http://www.backlog.jp/git-guide/stepup/stepup7_5.html	234
	C	https://www.google.co.jp/#q=git+branch+move[2]	131
5	A	https://www.google.co.jp/#q=git+rebase	6
	B	http://sota1235.com/blog/2015/03/19/git-rebase.html	70
	C	http://qiita.com/atskimura/items/a90dfa8bfc72e3657ef9	112

Table 2. Types and viewing time of reference pages by user

User	Articles[sec]	References[sec]	Search[sec]	QA[sec]	Total[sec]
A	128	77	39	0	244
B	820	674	659	5	2158
C	408	674	281	0	1363

6 Application

From the experiment for user activity module and preliminary experiments for other modules, we extended the method and implemented to reproduce manuals automatically on user activity.

6.1 Importance Score by Image Classification Module

Following four image classes are used in this paper.

1. branch : images about git branch
2. concept : images about git concept
3. decoration : ad banner and characters
4. CUI : images about command on CUI

Several variables are as follows:

- N_c^d : The number of images assigned to class c in page d
- N^c : The number of class
- q_c : The weight of class c for learning topic a

The importance of page d is defined as following function:

$$F_{\text{image}} = \sum_{i=0}^{N^c} q_i N_i^d \tag{1}$$

The images of branch and concept are important, but that of decoration is not so important. Thus, we set each q_c as follows:

- branch : 0.35
- concept : 0.4
- decoration : 0.0
- CUI : 0.25

6.2 Importance and kinds of pages by User Activity Module

Several variables about viewing time and the frequency for d as follows.

- t_i^d : i-th viewing time for d
- N^d : Times of viewing page d

Total viewing time is defined as follows:

$$T^d = \sum_{i=0}^{N^d} t_i^d \tag{2}$$

Threshold $t^{\text{threshold}}$ [sec] is defined and the importance and kinds of pages are obtained by the type of following cases between two types of viewing time and page.

- $T^d < t^{\text{threshold}}$: page d is not so important.
- $T^d \geq t^{\text{threshold}}$, $\max\{t_i^d\} < t^{\text{threshold}}$: The total time of page d is long but the each viewing time is short. e.g. the page d about articles for commands or search result pages.
- $\max\{t_i^d\} > t^{\text{threshold}}$: page d is important or d is difficult to read; the number of images in d is small.

Three module calculates the difficulties of g and the importance h of candidate pages d. The relation between interest V of user u and h, g. After that, semantic duplicate pages d is removed from page set D by Document similarity module.

Then, the method obtained existed candidate pages D' for learning topic a of user u. It can be reproduced manuals by combining sentences and images of d_i based on the structure of $d_i \in D'$

7 Conclusion

In this research, we propose the method to create manual automatically utilizing three types classification; document classification, image classification and user activity classification.

We have implemented the application based on the proposed method. In the experiment, our proposed method have been applied to the process of learning git as an example and we confirmed the efficiency for personal learning. Followings are involving further works.

- Adjust more suitable threshold considering users' personalities
- Sophisticate arrangements screen-shots to generate the manuals
- Analyze more complex behavior to generate detailed manuals considering a weakness of each learner.

References

1. Qiita, https://qiita.com/
2. Stack Overflow, http://stackoverflow.com/
3. S. Ishimaru et. al., "Smart Eyewear for Interaction and Activity Recognition", *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp.307-310, (2015)
4. K. Murakami et. al., "A Proposal of an Automatic Installation Manual Generation Method Using Operating Logs for Open Source Software", *Information Processing Society of Japan*, pp.926-939, (2008)
5. Git, https://git-scm.com/
6. JINS MEME Academic, https://jins-meme.com/en/
7. Q. Le et al., "Distributed Representations of Sentences and Documents", ICML, vo.l.14, pp.1188–1196, (2014)
8. Q. Le., "Building high-level features using large scale unsupervised learning", *In Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8595-8598, (2013)
9. K. Fukushima et al., "Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position", *Pattern Recognition*, Vol. 15, Issue 6, pp. 455-469(1982)
10. Learn Git Branching, http://learngitbranching.js.org/

Privacy-Utility Feature Selection as a tool in Private Data Classification

Mina Sheikhalishahi, Fabio Martinelli

Istituto di Informatica e Telematica, Consiglio Nazionale delle Ricerche, Pisa, Italy
{Mina.Sheikhalishahi, Fabio.Martinelli}@iit.cnr.it

Abstract. This paper presents a novel framework for privacy aware collaborative information sharing for data classification. Two data holders participating in this information sharing system, for global benefits are interested to model a classifier on whole dataset, if a certain amount of privacy is guaranteed. To address this issue, we propose a privacy mechanism approach based on privacy-utility feature selection, which by eliminating the most irrelevant set of features in terms of accuracy and privacy, guarantees the privacy requirements of data providers, whilst the data remain practically useful for classification. Due to the fact that the proposed trade-off metric is required to be exploited on whole dataset, secure weighted average protocol is utilized to protect information leakage in each site.

1 Introduction

Facing the new challenges brought by a continuous evolving Information Technology (IT) market, large companies and small-to-medium enterprises found in *Information Sharing* a valid instrument to improve their key performance indexes. Sharing data with partners, authorities for data collection and even competitors, may help in inferring additional intelligence through collaborative information analysis [11], [13]. Such an intelligence could be exploited to improve revenues, e.g. through best practice sharing [4], market basket analysis [12], or prevent loss coming from brand-new potential cyber-threats [6]. Other applications include analysis of medical data, provided by several hospitals and health centers for statistical analysis on patient records, useful, for example, to shape the causes and symptoms related to a new pathology [1].

Independently from the final goal, unfortunately information sharing brings issues and drawbacks which must be addressed. These issues are mainly related to the information privacy. Shared information may contain the sensitive information, which could be potentially harming the privacy of physical persons, such as employee records for business applications, or patient records for medical ones. Hence, the most desirable strategy is the one which enables data sharing in secure environment, such that it preserves the individual privacy requirement while at the same time the data are still practically useful.

In present study, we assume that two data providers are interested to model a classifier on shared data. For example, two hospitals are interested to use the result of classifiers on patients' records to identify the trends and patterns of diseases. The result on whole dataset brings the benefit for both parties to find the better treatment. However,

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_31

for privacy concerns, the hospitals are unwilling to disclose the patients' records, unless that a privacy level is satisfied [11]. To this end, we propose a privacy-aware feature selection framework which by secure removing the most *irrelevant* features, from both datasets, increases privacy gain while slightly modifies the data utility.

Generally, *feature selection* is based on the notion that a subset of features from the input can effectively describe the data [5]. This means that the information content can be obtained from the smaller number of variables which represent more discrimination information about the classes. On the other side, removing a set of variables increases the uncertainty of new dataset comparing to the original one, i.e. it increases the privacy gain. The rational behind it comes simply from the fact that each feature carries some information about data, such that by its removal the data become less indicative. However, the optimum set of features in terms of data efficiency are not necessarily equivalent to the best set of feature in terms of privacy gain. To address this issue, we propose an approach based on trade-off metric between *data utility* (classification accuracy) and *privacy gain*, when data is distributed between two parties.

A privacy-aware feature selection framework has been proposed in [9], which reports the efficiency of feature selection approach when privacy and utility are balanced in data publishing. However, in proposed framework only one data holder publishes securely her own dataset. In present study, we assume that two data holders are interested to share their own datasets to model a classifier on whole data. In the case that two parties are involved, it is required that some secure computation protocols be exploited. In present study, we utilize a *secure weighted average* protocol to find the proper subset of features, where the required privacy of each data provider is preserved, and the modified shared information sustain data utility for classification.

The rest of the paper is structured as follows. In Section 2 the related work on two concepts of privacy preserving and feature selection is presented. Section 3 presents *secure weighted average protocol* exploited in this study. Section 4 describes the problem statement and the proposed framework, detailing the secure computation of privacy and utility scores. Finally Section 5 briefly concludes proposing future research directions.

2 Related Work

Several work have been devoted to secure feature selection in multiparty data analysis framework. As a remarkable study, in [2], a secure distributed protocol is proposed which allows feature selection for multiple parties without revealing their own dataset. The proposed approach is based on *virtual dimension reduction* for selecting the subset of features in image processing. In this paper, the methodology is designed to privately select the most appropriate subset of features. However, differently from our approach, privacy gain does not come under consideration in feature selection. Moreover, the dimension reduction technique is exploited for unsupervised algorithms, e.g clustering, rather than classification. In [8], feature selection is combined with anonymization techniques to remove redundant features in order to publish a secure dataset. The result is evaluated through computing the accuracy of classification on original and sanitized datasets applying UCI benchmark datasets. The authors show that in some cases the accuracy even improves comparing to the original dataset. However, in this work, dif-

ferently from our approach, the dataset is centralized, and just one data holder is involved. In [9], a framework is proposed to select the few features which contain maximum discriminative information of classes. The classification accuracy and privacy gain are combined in feature selection process. However, the proposed approach, differently from our methodology, does not consider the case of privacy-aware feature selection in distributed parties.

To the best of our knowledge, it is among the very first work which incorporates the usefulness and the amount of privacy that a feature carries to shape optimum feature selection, as a service, when it is desired to model a classifier on whole data of two parties.

3 Secure Weighted Average Protocol (WAP)

Alice and *Bob* have (a_1, a_2) and (b_1, b_2), respectively. The two parties are interested to jointly compute $\frac{a_1+b_1}{a_2+b_2}$ in a secure way that each will know the final result without knowing the input of the other party.

In this study, we exploit the secure weighted average protocol proposed in [10] as what follows. Let (G, E, D, M) be an encryption scheme, where G is the function for generating public parameters, E and D are the encryption and decryption functions, and M is the message space, with the following properties:

- The encryption scheme (G, E, D) is semantically secure, i.e. an adversary gains no extra information from inspecting ciphertext.
- For all $m, \alpha \in M$, $m_1 \in E(x)$ means that $m_1^{\alpha} \in E(m\alpha)$, i.e. $E(m)$ denotes the set of ciphertexts can be obtained by the encryption of m.
- There exists a computable function f such that for all messages m_1 and m_2 we have $f(E(m_1), E(m_2)) = E(m_1 + m_2)$.

Keeping the above properties of our required probabilistic encryption system, the protocol of secure weighted average is presented as follows.

Assume that *Alice* sets up a probabilistic encryption scheme (G, E, D, M) where the parameters of G are public [10].

- (Step 1): *Alice* encrypts a_1 and a_2 and sends the encrypted values $a_1^{'} \in E(a_1)$ and $a_2^{'} \in E(a_2)$ to *Bob*.
- (Step 2): *Bob* computes a random message $m \in M$ and encrypts $m \cdot b_1$ and $m \cdot b_2$ to obtain $m_1' \in E(m \cdot b_1)$ and $m_2' \in E(m \cdot b_2)$. Then, *Bob* calculates $\mathcal{M}_1 = f(a_1^{'m}, m_1')$, $\mathcal{M}_2 = f(a_2^{'m}, m_2')$, and sends the result to *Alice*, where f is computed through multiplication.
- (Step 3): From the properties of probabilistic scheme (G, E, D), the following are obtained:
$$\mathcal{M}_1 = E(m \cdot a_1 + m \cdot b_1) \quad , \quad \mathcal{M}_2 = E(m \cdot a_2 + m \cdot b_2)$$
Hence, *Alice* is able to compute $m \cdot (a_1 + b_1)$ and $m \cdot (a_2 + b_2)$ and consequently $\frac{a_1+b_1}{a_2+b_2}$. *Alice* sends the final result to *Bob*.

Let us to show the result of this protocol as a function which gets as input two pairs of (a_1, a_2) and (b_1, b_2) and returns *the weighted average* of these pairs through the above protocol. We denote this function by *WAP* ($(a_1, a_2), (b_1, b_2)$).

4 Problem Statement and Proposed Framework

Let consider that two data holders are interested to share their own data to model a classifier. It is assumed that the data are distributed *horizontally* among parties. This means that each data holder involved in data sharing has information about all the features but for different collection of objects. In this scenario, the set of features applied to describe the records are known beforehand. Let $\mathcal{A} = \{A_1, A_2, \ldots, A_t\}$ be the set of t categorical features, all used to express each record of data, and the class labels come from the set $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$. Therefore, each record is a $t+1$ dimensional vector $z_i = (v_{i1}, v_{i2}, \ldots, v_{it}, C_i)$, where the first t components correspond to the features describing the record z_i, i.e. $v_{ij} \in A_j$, for $1 \leq j \leq t$, and the last component presents the class label of z_i, i.e $C_i \in \mathcal{C}$. For privacy issues, the data holders accept to share their own dataset only if a minimum amount of privacy is guaranteed.

To obtain the optimal subset of features, say $\{A_1^*, A_2^*, \ldots, A_p^*\} \subseteq \mathcal{A}$, first two parties set together the minimum amount of *privacy gain* denoted by θ, desired to be preserved through removing a subset of features. Then, *privacy* and *utility* scores for each feature, on whole data, are computed with the use of *secure weighted average protocol*. Finally, after secure computation of feature utility and privacy scores, the trade-off score for each feature is computed to obtain the optimum subset of features. The feature with the minimum privacy-utility trade-off score is the first candidate to be removed. If after removing the first feature, the privacy requirements of both data holders satisfies (i.e. *privacy gain* $> \theta$), then that specific feature is removed and the datasets are published; Otherwise, the next feature in terms of minimizing the privacy-utility trade-off score is removed. In the case that by removing half of the features the privacy requirements of parties are not satisfied, the data holders are required to refine the restriction of privacy gain threshold.

4.1 Secure Utility Score Computation

The aim of *feature selection* in data mining algorithms is to obtain an *optimal* set of features such that it contains all *relevant* features which are not *redundant* in terms of identifying the class labels of a dataset [5]. There are many potential benefits of feature selection, spanning from facilitating data visualization and understanding, reducing the measurement and storage requirements, reducing training and utilization times, to defying the curse of dimensionality to improve prediction performance [7]. *Feature ranking* as a well-known feature selection approach, creates a scoring function, say $\upsilon(A)$, computed from the impact of feature A in discriminating class labels. Generally, a high score is indicative of a valuable feature. In present study, we apply a well-known feature ranking technique based on *Mutual Information* [5] to score the features based on their *utility*.

Formally, let $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$ be the class labels of a dataset D, and $A = \{v_1, v_2, \ldots, v_{|A|}\}$ be the set of values of the feature A. Then, the *Shannon entropy* of variable $\mathcal{C}$ is defined as follows:

$$H(\mathcal{C}) = -\sum_{C_i \in \mathcal{C}} p(C_i) \log(p(C_i))$$

where $p(C_i)$ is the number of records in D labeled C_i divided by the total number of records in D. The *conditional entropy* of the output $\mathcal{C}$ to variable A is given by:

$$H(\mathcal{C}|A) = -\sum_{v_k \in A} \sum_{C_i \in \mathcal{C}} p(v_k \cdot A, C_i) \log(p(C_i|v_k \cdot A))$$

where $p(v_k \cdot A, C_i)$ represents the number of elements respecting k'th value of A and having the class label C_i divided by the whole number of records in D. The decrease in uncertainty of the output $\mathcal{C}$ observing feature A, denoted by function $\upsilon(A)$, is computed as $\upsilon(A) = H(\mathcal{C}) - H(\mathcal{C}|A)$, where $H(\mathcal{C})$ and $H(\mathcal{C}|A)$ represent the *Shannon entropy* and *conditional entropy*, respectively. We call $\upsilon(A)$ the *utility score* of feature A. The feature with the lower score is the one desired to be first removed, since it has the minimum relevance in identifying the class labels of the records. Algorithm 1 details the process of secure *utility score* computation between two parties.

Algorithm 1: *utility.score()* : Secure Utility Score Computation

```
1  initialization;
2  for 1 ≤ j ≤ t do
3      for 1 ≤ k ≤ |A_j| do
4          for 1 ≤ i ≤ m do
5              Alice: a_1 ← number of records in D_a respecting v_k · A_j and C_i
6              Alice: a_2 ← total number of elements in D_a
7              Alice: a'_1 ← number of records in D_a with class label C_i that respect
                   v_k · A_j
8              Alice: a'_2 ← number of records in D_a with class label C_i
9              Bob: b_1 ← number of records in D_b respecting v_k · A_j and C_i
10             Bob: b_2 ← total number of elements in D_b
11             Bob: b'_1 ← number of records in D_b with class label C_i that respect
                   v_k · A_j
12             Bob: b'_2 ← number of records in D_b with class label C_i
13             p(v_k · A_j, C_i) ← WAP((a_1, a_2), (b_1, b_2))
14             p(C_i|v_k · A_j) ← WAP((a'_1, a'_2), (b'_1, b'_2))
15             υ(A_j) ← υ(A_j) − p(v_k · A_j, C_i) log(p(C_i|v_k · A_j))
16         end
17         return υ(A_j)
18     end
19 end
```

Theorem 1. *Algorithm 1 reveals nothing to the other party, except* feature utility *scores on whole data.*

Proof. The only communication between two parties occur at lines 13 and 14, which is a call to secure weighted average computation protocol, proven to be secure in [10]. □

4.2 Secure Privacy Score Computation

Data privacy is quantified as the degree of uncertainty that the original data can be inferred from the sanitized one [3]. Generally, reducing the information which a dataset carries will increase privacy gain. Hence, removing a feature from a dataset can be considered as a tool which increases this uncertainty. We compute the *privacy score* [11] resulting from removing feature A_j from original dataset D and obtaining the sanitized dataset $D - \{A_j\}$, denoted by $\rho(D, D - \{A_j\})$, as follows:

$$\rho(D, D/\{A_j\}) = -\sum_{s=1}^{t}\sum_{k=1}^{|A_s|}(p(v_k \cdot A_s) \cdot \log(p(v_k \cdot A_s) = -\sum_{s\neq j}^{t}\sum_{k=1}^{|A_s|} p(v_k \cdot A_s) \cdot \log(p(v_k \cdot A_s))$$

where $|A_s|$ is the number of values for the s'th attribute, $p(v_k \cdot A_s)$ denotes the number of records that respects k'th value of s'th attribute divided by the number of records. For the sake of simplicity, when the privacy gain score is computed for one specific attribute, say A_j, we denote the privacy score of feature A_j as $\rho(A_j)$ instead of $\rho(D, D - \{A_j\})$. Algorithm 2 details the process of secure privacy score computation between two parties.

Algorithm 2: *privacy.score()*: Secure Privacy Score Computation

```
1  initialization;
2  for 1 ≤ j ≤ t do
3      for 1 ≤ j' ≤ t, j' ≠ j do
4          for 1 ≤ k ≤ |A_j| do
5              for 1 ≤ k' ≤ |A_j'| do
6                  Alice: a_1 ← number of records in D_a respecting v_k · A_j
7                  Alice: a_2 ← total number of elements in D_a
8                  Alice: a' ← number of records in D_a − {A_j} which respect v_k' · A_j'
9                  Bob: b_1 ← number of records in D_b respecting v_k · A_j
10                 Bob: b_2 ← total number of elements in D_b
11                 Bob: b' ← number of records in D_b − {A_j} which respect v_k' · A_j'
12                 p(v_k · A_j) ← WAP((a_1, a_2), (b_1, b_2))
13                 p'(v_k' · A_j') ← WAP((a'_1, a_2), (b'_1, b_2))
14                 ρ(A_j) ←
                   −(p(v_k · A_j) log(p(v_k · A_j)) − p'(v_k' · A_j') log(p'(v_k' · A_j')))
15             end
16         end
17     end
18     return ρ(A_j)
19 end
```

Theorem 2. *Algorithm 2 reveals nothing to other party except the privacy scores of features.*

Proof. The only communication between *Alice* and *Bob* occur at lines 12 and 13, which is a call to secure weighted average protocol, proven to be secure in [10]. □

4.3 Privacy-Utility Feature Selection

A simple expression matching trade-off score properties, which gives the same weight to feature privacy gain and feature utility,could be defined as $\tau(\upsilon(A),\rho(A)) = \frac{1}{2}(\upsilon(A)+\rho(A))$, where $\rho(A)$ and $\upsilon(A)$ are the privacy and utility scores of feature A. From the proposed metric, the feature which gets the minimum score is the best candidate to be removed. Algorithm 3 details the process of secure privacy-utility score computation.

Algorithm 3: *privacy.utility()*: Secure Privacy-Utility Score Computation

Data: *Alice* and *Bob* have statistical information of $\{A_1, A_2, \ldots, A_k\}$
Result: The feature respecting minimum privacy-utility score

```
1  initialization;
2  A* = A_1
3  for 1 ≤ j ≤ t do
4  |   υ(A_j) ← utility.score(A_j)
5  |   ρ(A_j) ← privacy.score(A_j)
6  |   τ(A_j) = 1/2 (υ(A_j) + ρ(A_j))
7  |   if τ(A*) > τ(A_j) then
8  |   |   A* ← A_j
9  |   end
10 end
11 return A*
```

Algorithm 3 is secure, since the only communications between *Alice* and *Bob* occur at lines 4 and 5, and they have proven to be secure in Theorems 1 and 2.

After executing Algorithm 3, both *Alice* and *Bob* compute the *privacy gain* on their original dataset, and sanitized dataset resulted from removing *irrelevant* features, say $\{A_{j1}, A_{j2}, \ldots, A_{jl}\}$, as the following:

$$\rho(D, D-\{A_{j1}, A_{j2}, \ldots, A_{jl}\}) = -\sum_{s=1}^{t}\sum_{k=1}^{|A_s|}(p(v_k \cdot A_s)\cdot\log(p(v_k \cdot A_s)) - \sum_{s=1, s\notin\{j1,\ldots,jl\}}^{t}\sum_{k=1}^{|A_s|}(p(v_k \cdot A_s)\cdot\log(p(v_k \cdot A_s))$$

If in both sides the *privacy gain* $\rho(D, D-\{A_{j1}, A_{j2}, \ldots, A_{jl}\})$ is higher than θ, *Alice* and *Bob* publish the sanitized datasets; Otherwise, the next feature satisfying the minimum privacy-utility score is found through Algorithm 3. The process continues till both parties reach the required privacy threshold.

5 Conclusion

Feature selection is a tool in machine learning which is used to identify a subset of features that mostly discriminate the information about classes. To this end, to evaluate the usefulness of the feature, generally the classification accuracy is applied. Such a metric does not take into account the privacy gain of published dataset. In this paper, we applied feature selection and privacy gain as an ensemble tool to find the best set of features in terms of privacy-utility trade-off in distributed data sharing architecture. The proposed approach, with the use of secure weighted average protocol, securely removes the set of irrelevant features to shape a tool for modeling a classifier on shared data.

In the future directions, we plan to generalize the proposed approach to a framework respecting different trade-off metrics of different privacy and utility metrics. Moreover, we aim to exploit the proposed approach on real benchmark datasets to evaluate a real case-study. Also, we plan to solve the same issue, i.e. finding the optimum subset of feature in terms of privacy-utility trade-off, in the case that the data are partitioned vertically among different data holders.

Acknowledgment

This work was partially supported by the H2020 EU funded project NeCS [GA #675320] and by the H2020 EU funded project C3ISP [GA #700294].

References

1. Artoisenet, C., Roland, M., Closon, M.: Health networks: actors, professional relationships, and controversies. In: Collaborative Patient Centred eHealth. vol. 141. IOSPress (2013)
2. Banerjee, M., Chakravarty, S.: Privacy preserving feature selection for distributed data using virtual dimension. In: Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM. pp. 2281–2284 (2011)
3. Bertino, E., Lin, D., Jiang, W.: A survey of quantification of privacy preserving data mining algorithms. In: Privacy-Preserving Data Mining, vol. 34, pp. 183–205. Springer US (2008)
4. Bogan, C.E., English, M.J.: Benchmarking for best practices : winning through innovative adaptation. New York : McGraw-Hill (1994)
5. Chandrashekar, G., Sahin, F.: A survey on feature selection methods. Comput. Electr. Eng. 40(1), 16–28 (Jan 2014)
6. Faiella, M.F., Marra, A.L., Martinelli, F., Francesco, Saracino, A., Sheikhalishahi, M.: A distributed framework for collaborative and dynamic analysis of android malware. In: 25th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP), St. Petersburg, Russia (2017)
7. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning 3, 1157–1182 (2003)
8. Jafer, Y., Matwin, S., Sokolova, M.: Task oriented privacy preserving data publishing using feature selection. In: Advances in Artificial Intelligence - 27th Canadian Conference on Artificial Intelligence. pp. 143–154 (2014)
9. Jafer, Y., Matwin, S., Sokolova, M.: A framework for a privacy-aware feature selection evaluation measure. In: 13th Annual Conference on Privacy, Security and Trust, PST 2015, Izmir, Turkey, July 21-23, 2015. pp. 62–69 (2015)
10. Jha, S., Kruger, L., McDaniel, P.: Privacy Preserving Clustering, pp. 397–417. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)
11. Martinelli, F., Saracino, A., Sheikhalishahi, M.: Modeling privacy aware information sharing systems: A formal and general approach. In: 15th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (2016)
12. Oliveira, S.R.M., Zaïane, O.R.: Privacy preserving frequent itemset mining. In: Proceedings of the IEEE International Conference on Privacy, Security and Data Mining - Volume 14. pp. 43–54. CRPIT '14 (2002)
13. Sheikhalishahi, M., Mejri, M., Tawbi, N., Martinelli, F.: Privacy-aware data sharing in a tree-based categorical clustering algorithm. In: Foundations and Practice of Security - 9th International Symposium, FPS 2016, Québec City, QC, Canada. pp. 161–178 (2016)

Inference of Channel Priorities for Asynchronous Communication

Nathanaël Sensfelder[1], Aurélie Hurault[1] and Philippe Quéinnec[1]

IRIT - Université de Toulouse, 2 rue Camichel, F-31000 Toulouse, France
http://www.irit.fr

Abstract. In distributed systems, the order in which the messages are received by the processes is crucial to ensure the expected behavior. This paper presents a communication model which allows for restrictions on the deliveries of a channel depending on the availability of messages in other channels. This corresponds to prioritizing some channels over others. It relies on a framework able to verify if a given system satisfies a user defined LTL (Linear Temporal Logic) property with different priorities. We also propose to automatically infer the channel priorities so that the system does not infringe on this temporal property.

1 Introduction

In an asynchronous environment, the delivery of messages is by essence nondeterministic. This is bound to cause complications, notably with software testing, making the possibility of a formal verification of the system's correctness highly desirable. Many communication models impose additional constraints on the delivery of a message, such as First-In-First-Out or causally ordered communication. Those communication models belong in two classes: the generic models, which are defined independently of the system that uses them; and the applicative models, whose constraints refer to the system's components (such as, in our case, its channels). The latter require the communication model to be updated whenever the definition of the system is changed. This makes it impractical for non-trivial systems if the constraints are not automatically inferred.

Consider a distributed system composed of peer exchanging messages over predefined channels. Following the Calculus of Communicating Systems (CCS) syntax, the sending of a message on a channel is indicated by a '!', the reception by a '?', and a peer internal action by 'τ'. Using a purely asynchronous communication model (unlike CCS) in the system described in Figure 1 fails to prevent the last peer from going into a forbidden state (noted $\bot$), for instance via the sequence of events: $a! \cdot b! \cdot b? \cdot c! \cdot c?$. The same issue occurs using point-to-point communication with FIFO ordering (first in first out, i.e. a queue between each

[1] This work was partially supported by project PARDI ANR-16-CE25-0006. An extended version of the paper is available at http://vacs.enseeiht.fr/dcai17-long.pdf.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_32

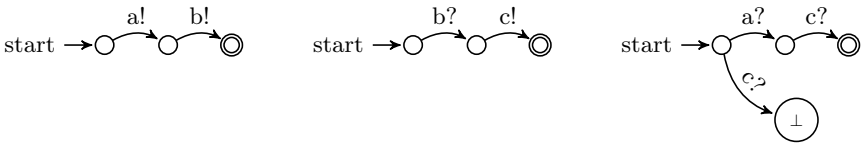


Fig. 1. Three Peers Interacting With Three Channels a, b, c. $a!$ is a send event, $a?$ a receive event, communication is asynchronous.

couple of peers), whereas using a causal communication model would avoid the problematic executions. However, using that causal communication model comes at a cost, making the use of an applicative solution worth considering. The issue comes from a process receiving from channel c instead of channel a, despite both being available. In this case, channel a should have a higher priority than c.

Another use of the channel priorities is found when trying to reduce the nondeterminism of a system, even when all possible executions are valid, should certain executions be preferable to other. A classic example is abortion messages. If the communication model allows the system to take other messages over the abortion one, this results in a seemingly unresponsive behavior to abortion or presents security issues.

The outline of this paper is the following. Section 2 presents the framework for the verification of asynchronous communication and precisely defines what priorities mean. Section 3 describes the inference algorithm which uses the framework to discover the necessary priorities in order to ensure the correct behavior of a system. Section 4 illustrates our approach with an example. Section 5 gives an overview of other approaches for ordering communication interactions and Section 6 provides perspectives and final remarks.

2 Verification of Asynchronous Communication with Priorities

2.1 The Framework

Our objective is to tell if a system, composed of peers and of a communication model, verifies a correctness property given by the designer. The peers asynchronously interact through channels, and the communication model decides the delivery of messages (e.g. in what order the messages are available). Following [6], we have built a framework and an automated toolchain, based on TLA$^+$ [13] and its tools, that enables to check an LTL (Linear Temporal Logic) property on a system. As the communication models are TLA$^+$ modules which are composed with the peers, the framework allows to easily verify how a set of peers interacts with several models, or if the specified parameters are sufficient to validate the expected property.

In the present case, the communication model with channel priorities is specified by a set of BLOCKS constraints. Constraint (A BLOCKS b) means: if any of

the channels in the set A contains a message, reception on channel b is disabled. Note that the BLOCKS constraints do not necessarily form a partial order on channels.

The framework is accessible online at http://vacs.enseeiht.fr/. It can be used both to check if a system is correct (for a selection of predefined LTL properties), and to discover the necessary priorities to make it correct (if possible).

2.2 Formalization

This section presents the formal definitions upon which the framework is built, and gives the precise definition of the BLOCKS constraint. Classically, the system resulting of the composition of peers and a communication model is defined as a labelled transition system, and an execution as a sequence of states. *Net* is an abstraction of the messages in transit. The order of delivery by the communication model is based on the channel priorities defined by BLOCKS.

Definition 1 (Composed System). *A system is a quintuplet* $(States, Init, Labels, Relation, Channels)$ *where*

- $Labels \subseteq (Channels \times \{\text{“?”}, \text{“!”}\}) \cup \{\tau\}$. $c!$ *is interpreted as the sending of a message on channel* c*, and* $c?$ *as the reception of a message from channel* c*, and* τ *is an internal action;*
- $Relation \subseteq (States \times Labels \times States)$ *is the transition relation.*
- $Init \subseteq States$ *are the initial states.*

Definition 2 (Execution). *The set of all possible executions is the set of all finite or infinite sequences of states where consecutive states conform to the transition relation, and such that a message is received at most once and this reception is preceded by a send.*

Definition 3 (Network). *At any point of an execution, Net is the set of channels where at least one message is in transit.*

Definition 4 (Disabled reception). *Reception on a channel* c *is disabled at a given point of an execution if it does not occur at that point.*

Channel priorities are a set of static constraints that forces specific receptions to be *disabled*, depending on the values of *Net* and the *Channels* those transitions relate to. (B BLOCKS c) means that whenever at least one message is present on any channel of B, then the reception on c is *disabled*.

Definition 5 (BLOCKS). *A system Sys parameterized by the C* BLOCKS *constraints respects:*

$$Sys, C \models \forall B \subseteq Channels, \forall c \in Channels,$$
$$(B \text{ BLOCKS } c) \in C \Rightarrow \Box(\forall b \in B, b \in Net \Rightarrow disabled(c?))$$

3 Inferring Channel Priorities

When inferring the channel priorities, the objective is, given a system Sys and a property P, to find all the BLOCKS constraint sets C such that $Sys, C \models P$. We define an analyzer which asks the framework whether a set of constraints applied to Sys satisfies P. The analyzer infers new constraints using the counter-example given by the framework when the property is not verified with the proposed constraint set. This yields to the generation of new sets of constraints, called candidates, built by adding new constraints to the current candidate.

3.1 Introducing New Constraint Types

The candidates are built in a incremental manner. Because of this, we have to ensure that any constraint added to a candidate does not invalidate the need for the existing ones. Indeed, adding a new BLOCKS constraint may cause a previously taken reception transition to become *disabled* without notice. This is resolved by the use of two other constraints types, exclusively used during the inference process: the ALLOWED constraints, and the BLOCKED constraints.

The (a ALLOWED b) constraint indicates that the building of the candidate has allowed the exploration of states that would be unreachable should channel a BLOCKS channel b. A constraint (a BLOCKS b) is thus prevented from being added to this candidate.

The (B BLOCKED c) constraint, on the other hand, is used to convey that at least one channel in the B set BLOCKS the channel c. Not specifying which channel does the actual blocking, like we would have to when using BLOCKS constraints, lets the incremental inferring process add (d ALLOWED c) with $d \in B$ to the candidate as long as, for any set of constraints C:

$$\forall B \subseteq Channels, \forall c \in Channels, (B \text{ BLOCKED } c) \in C \Rightarrow \\ \exists b \in B, (b \text{ ALLOWED } c) \notin C$$

As we don't know which element(s) of B in (B BLOCKED c) are blocking c, there are now three possibilities when a peer attempts reception on c. Either c is BLOCKED by all the channels in the blocking set and the reception is disabled, or c is ALLOWED by all of the channels in Net and the reception is possible, or the situation is *ambiguous* and it is yet unknown if the reception is possible.

Definition 6 (Ambiguity). *At a given state,* available *is the set of all channels that can be received from by the peer at that state.* Ambiguity *is the subset of* available *channels which are not* ALLOWED *by at least one of channels holding at least one message.*

$$available(peer) \triangleq \{channel \in Channels \mid \text{ENABLED}^1\ receive(peer, channel)\}$$
$$Ambiguity(peer) \triangleq \\ \{c \in available(peer) \mid \exists\ ch \in Net, (c \neq ch) \wedge \neg(ch \text{ ALLOWED } c)\}$$

[1] In TLA$^+$, ENABLED *Action* is true in a state if the action is possible, meaning there is a successive state reachable with *Action*. The action $receive(p, c)$ is enabled in a state if c is not blocked by a BLOCKED constraint at that point.

3.2 The Analyzer

The analyzer handles sets of candidates, each of which is a set of ALLOWED and BLOCKED constraints. It starts with a single candidate without any constraint. Information on the system is gathered by choosing a candidate, setting it as the constraint set of the communication model, and then asking the framework to report. The framework either declares that the targeted property is validated, or gives its report of *ambiguous* states. This report is $\langle net, channels \rangle$, the current value of Net and the *Ambiguity* set. When the framework reports that the expected property is verified, the candidate is added to the solutions. Otherwise, if there is no *ambiguous* state, the candidate is rejected; if *ambiguous* states are reported, the analyzer replaces the candidate by its children, each of which is generated using the following function where *chosen* is any subset of *channels*:

$$\begin{aligned} &update(Candidate, \langle net, channels \rangle, chosen) \triangleq \\ &\quad Candidate \\ &\cup \{((net \setminus \{v\}) \text{ BLOCKED } v) \mid v \in channels \setminus chosen\} \\ &\cup \{(c \text{ ALLOWED } v) \mid v \in chosen \wedge c \in net\} \end{aligned}$$

Every subset of *channels* generates its own child (including $\emptyset$). The ALLOWED constraints make sure that the elements of *chosen* are not BLOCKED. The elements of *channels* that are not *chosen* are BLOCKED, so there is no longer any ambiguity. Obviously, inconsistent children (e.g. a child with both ({a} BLOCKED b) and (a ALLOWED b)) are discarded.

The inference process offers two variations of the analyzer. The Pessimistic Priority Finder find all the possible solutions. When studying a candidate, the framework stops and reports at each ambiguous state, and the analyzer explores all the children of this candidate. The Optimistic Priority Finder sacrifices exhaustiveness for performance. In that case, the framework pursues the verification until it finds an invalidation of the desired property (in which case another candidate is generated), or it validates the candidate.

4 Example: A Client-Controller-Application System

A system is composed of three peers: a client, a controller and an application (Figure 2). The client interacts with the controller to get the authorization to access the application, then interacts with the application which has been started when needed by the controller. More precisely, the client sends a *login* to the controller which can *accept* of *reject* it. If accepted, the client can send *upload* messages to the application. This controller starts the application (message *start*) when it accepts a client, and signals it to *end* when the client *logout*s.

Several problems occur if the messages are arbitrarily delivered. Among others, the application must consume *start* before processing the messages *upload* whereas they come from different peers; *end* must not be consumed when there are pending *upload*s; when the client *logout*s then *login*s again, *start* must not overtake any of the messages from the previous round (notably *upload* or *end*)...

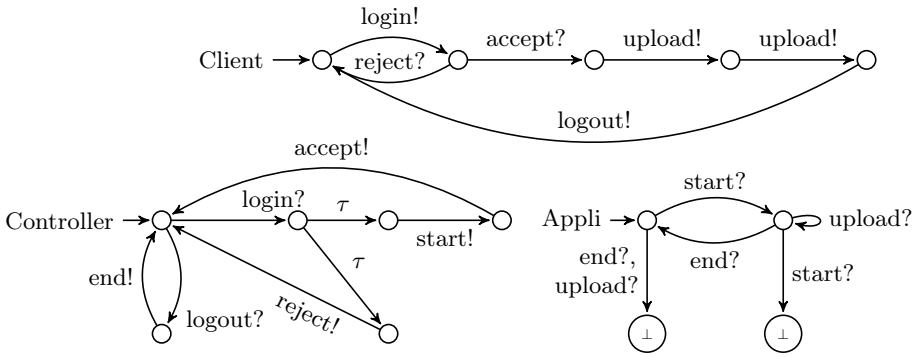


Fig. 2. A Client-Server System. The server is split into a controller (bottom left) and an application (bottom right). The controller accepts or rejects the client (top), and starts/ends the application when needed.

To avoid deadlock and $\bot$ states, specifying and manually verifying a correct ordering of messages is not easy. Our framework automatizes the verification and the inference algorithm discovers the seven possible solutions, among others:

($\{start\}$ BLOCKS $upload$)	($\{start\}$ BLOCKS $accept$)	($\{start\}$ BLOCKS $accept$)
($\{upload\}$ BLOCKS end)	($\{upload\}$ BLOCKS end)	($\{upload\}$ BLOCKS $logout$)
($\{logout, end\}$ BLOCKS $login$)	($\{logout\}$ BLOCKS $login$)	($\{logout, end\}$ BLOCKS $login$)
Distinct states: 298	($\{end\}$ BLOCKS $start$)	**Distinct states:** 99
	Distinct states: 465	

Observe that the solutions present a large scattering in the number of states, meaning that some solutions allow more executions than others. Note also that the client is not explicitly waiting for the application to progress: the message priorities ensure that the application does not lag behind too much.

5 Related Work

Generic ordering. Generic ordered delivery, such as FIFO or causal delivery, has been studied in the context of distributed algorithms. Asynchronous communication models in distributed systems have been studied in [12] (notion of ordering paradigm), [5] (notion of distributed computation classes), or [7] (formal description and hierarchy of several asynchronous models). Implementations of the basic models using histories or clocks are explained in classic textbooks [10, 12, 15]. These works deal with generic orderings, solely based on the behavior of the system (e.g. relative order of the events) but they are not application-defined.

Applicative Priority on Specific Transitions. In [4], a priority operator *prisum* is added to CCS, allowing for the expression of preference between two actions. Its semantics is similar to the extended constraints presented in section 6, as an enabled action forbids any action with a strictly lower priority. A difference is that the prisum relation is defined at state level, making it possible

for preference between two actions to change over the course of the execution, something that cannot happen with BLOCKS constraints. Conversely, expressing $(\{a\}$ BLOCKS $b), (\{b\}$ BLOCKS $a)$ using prisum does not appear to be possible.

The behavior of the ALT construct in the Occam programming language [11] lets its users give a list of channels to receive from, establishing a priority relation between them according to their location in the list. While this can easily be translated to BLOCKS constraints, with channel being blocked by all of those that surpass its priority in the ALT construct, we again face priorities that are established for (and at) a specific state.

Priorities have been introduced in Petri Nets. In [3], priorities are statically associated to couples of transitions. In interleaving semantics, a transition is enabled at a marking when no transition with higher priority is enabled at that marking. However, concurrent semantics cannot be naturally defined with causal partial order. Dynamic priorities are introduced in [2]. Priorities are a relation between couples of transitions, for each possible marking. The objective is to reduce the number of enabled transitions at a marking, in order to reduce the size of the reachability graph without affecting the truth of the studied property.

Applicative Priorities on Labels. Another point of view is to assign priorities to labels or message identities, as we have done in this paper. [1] uses this to provide an interrupt mechanism in process algebra. Priorities are associated to labels and form a partial order. Its semantics is defined by rules such that $a + b = a$ if $a > b$. However, this is not as easy as stated when labels can be masked and composability is lost if care is not taken. A thorough exploration of priority in process algebras with synchronous communication is done in [9]. The authors distinguish operational semantics and behavioral congruences (for compositional reasoning). Priorities are associated to labels, and are used only in a synchronous communication event. [14] is an extension to a broadcasting calculus with priority. Priorities are associated to processes and to send/receive events. A process can only receive messages with a higher priority than its own.

Controller synthesis. One difference with all the previous work is that our work not only defines priorities on channels and offers a framework to check if a temporal property is verified by the system, but we also provide an inference process to find all the solutions (if any) which guarantee that a given propriety is satisfied. This is reminiscent of controller synthesis [8]. We differ from this approach as we are not building a synchronizer based on the temporal property of interest, and several incomparable solutions are possible.

6 Generalization and Conclusion

On the whole, channel priorities are easy to use when the priorities are automatically inferred. Thanks to inference, fine grain interactions do not have to be specified when developing a system, and application-specific protocols are derived from the system requirements. This allows to focus on the architecture of the system, and to postpone protocol considerations until deployment.

The next step is to extend the constraints to all the action types: *send*, *receive*, and *tau*, all having priorities over one another. This can be achieved by making the channel priority constraint types work with *Actions* instead of *Channels*, an *Action* being $c?$ (receive from channel c), $c!$ (send on c) or τ. Interestingly enough, this does not cause any change to the inference analyzer and the framework is easily altered to take those new constraints into account.

This extension allows to set a constraint such as ($\{tick?\}$ BLOCKS $alarm!$), to make a peer report an alarm if and only if it is unable to receive an expected message. It also allows to express fairness constraint: for instance a ($\{a?\}$ BLOCKS τ) with τ-stuttering ensures that the peer does not get stuck in a τ loop if a reception on a is possible. This also allows a cancellation or abortion action to be easily described for any kind of action.

References

1. Jos C. M. Baeten, Jan A. Bergstra, and Jan Willem Klop. Syntax and defining equations for an interrupt mechanism in process algebra. *Fundamenta Informaticae*, IX:127–168, 1986.
2. Falko Bause. Analysis of Petri nets with a dynamic priority method. In *18th International Conference on Application and Theory of Petri Nets ICATPN'97*, volume 1248 of *Lecture Notes in Computer Science*, pages 215–234. Springer, 1997.
3. Eike Best and Maciej Koutny. Petri net semantics of priority systems. *Theoretical Computer Science*, 96(1):175–174, 1992.
4. Juanito Camilleri and Glynn Winskel. CCS with priority choice. *Information and Computation*, 116(1):26–37, 1995.
5. Bernadette Charron-Bost, Friedemann Mattern, and Gerard Tel. Synchronous, asynchronous, and causally ordered communication. *Distributed Computing*, 9(4):173–191, February 1996.
6. Florent Chevrou, Aurélie Hurault, and Philippe Quéinnec. Automated verification of asynchronous communicating systems with TLA+. *Electronic Communications of the EASST (special issue AVOCS'15)*, 72:1–15, 2015.
7. Florent Chevrou, Aurélie Hurault, and Philippe Quéinnec. On the diversity of asynchronous communication. *Formal Aspects of Computing*, 28(5):847–879, 2016.
8. Edmund M. Clarke and E. Allen Emerson. Design and synthesis of synchronization skeletons using branching-time temporal logic. In *Logics of Programs*, volume 131 of *Lecture Notes in Computer Science*, pages 52–71. Springer, 1981.
9. Rance Cleaveland and Matthew Hennessy. Priorities in process algebras. *Information and Computation*, 87(1/2):58–77, 1990.
10. George Coulouris, Jean Dollimore, and Tim Kindberg. *Distributed Systems: concepts and design*. Addison Wesley, second edition, 1994.
11. M. Elizabeth C. Hull. Occam - A programming language for multiprocessor systems. *Computer Languages*, 12(1):27–37, 1987.
12. Ajay D. Kshemkalyani and Mukesh Singhal. *Distributed Computing: Principles, Algorithms, and Systems*. Cambridge University Press, March 2011.
13. Leslie Lamport. *Specifying Systems*. Addison Wesley, 2003.
14. K. V. S. Prasad. A calculus of broadcasting systems. *Science of Computer Programming*, 25(2-3):285–327, 1995.
15. Michel Raynal. *Distributed Algorithms for Message-Passing Systems*. Springer, 2013.

Trends in Gravitational Search Algorithm

P. B. de Moura Oliveira[1], Josenalde Oliveira[1,2] and José Boaventura Cunha[1]

[1]INESC TEC – INESC Technology and Science,
Department of Engineering, School of Sciences and Technology,
5001–801 Vila Real, Portugal
[2]Agricultural School of Jundiaí- Federal University of Rio Grande do Norte, UFRN,
59280-000 Macaíba, RN, Brasil

Abstract. The gravitational search algorithm (GSA) is reviewed, by presenting a tutorial analysis of its key issues. As any other metaheuristic, GSA requires the selection of some heuristic parameters. One parameter which is crucial in regulating the exploratory capabilities of this algorithm is the gravitational constant. An analysis regarding this parameter selection is presented and a heuristic rule proposed for this purpose. The GSA performance is compared both with a hybridization with particle swarm optimization (PSO) and standard PSO. Preliminary simulation results are presented considering simple continuous functions optimization examples.

Keywords: Gravitational search algorithm, Particle Swarm Optimization.

1 Introduction

The gravitational search algorithm (GSA) is a physical inspired metaheuristic, which since its proposal by [1] has been growing in popularity due to the successful applications in several problem solving domains. Examples of practical applications are reported in control engineering [2,6], image segmentation [3,14] feature selection [5], system modelling [7], energy conversion and management [4,8] robotic path planning [12]. Significant research efforts have been devoted to develop GSA variations, extensions and hybrids with other metaheuristics, such as the following cases [9,10,11,12,13,14]. Despite these GSA refined versions most applications follow the pioneering algorithm proposed by [1], using some of the original heuristic settings independently of the problem to be solved. This can denote that either this metaheuristic is quite robust or that some of its key issues are badly understood.

The paper contributions are the following: i) The GSA algorithm is revisited by reviewing selected technique refinements. Thus, a tutorial approach is adopted and an attempt is made to present some GSA insights, in order to promote its easy perception and potential use; ii) Regarding the hybridization with other metaheuristics, the issues addressed here are focused into hybrids with the particle swarm optimization (PSO)

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_33

algorithm denoted PSOGSA; iii) A simple heuristic rule is proposed to select the initial value of the gravitational constant. Simulation results are presented which clearly show this parameter crucial importance in the GSA performance. Comparison results among GSA, PSOGSA and PSO algorithm are presented.

The rest of the paper is organized as follows: section two reviews the GSA algorithm presenting basic and selected topics, including hybridization aspects with PSO. Section three presents some simulation results and respective analysis and finally section four presents the conclusion and outlines future work.

2 Gravitational Search Algorithm: Basic and Selected Topics

Considering two agents moving in a search space, with masses represented by M_1 and M_2, according to the gravitational law defined by Newton the gravitational force, F, among these agents separated by a distance, R, can be represented by:

$$F = G\frac{M_1 M_2}{R^2} \tag{1}$$

with G representing the gravitational constant. Another well-known Newton law is the following:

$$F = Ma \Leftrightarrow a = \frac{F}{M} \tag{2}$$

relating force, F, with acceleration, a, through mass, M. Considering the current iteration represented by t, and two d-dimensional agents represented by i and j, with corresponding positions in the search space represented respectively by x_i and x_j the force among them was defined by [1] as:

$$F_{ij}^{d} = G(t)\frac{M_i(t)M_j(t)}{R_{ij}(t)+\varepsilon}(x_j^d(t) - x_i^d(t)) \tag{3}$$

with $\varepsilon>0$ representing a small constant and R the Euclidean distance. The insertion of ε is necessary to avoid division by zero, which otherwise occur for agents in the same exact location. Note that in (3) R is used instead of R^2 as stated in Newton law (1). This change originate doubt about the complete Newton gravity law incorporation into the GSA, as it is not an exact replica [15]. However this change was proposed by [1] based on experimental tests indicating that GSA performs better using R rather than using R^2. The gravitational constant G, in (3) was originally proposed as follows:

$$G(t) = G_0 \exp\left(-\alpha\frac{t}{t_{\max}}\right) \tag{4}$$

with G_0 representing the initial gravitational constant parameter, $\alpha>0$ a decay ratio parameter and t_{max} the total number of iterations. In the original GSA algorithm [1]

the following settings were proposed: G_0=100 and α=20. So, for each agent *i*, moving in a *d*-dimensional space, the total force exerted in *i*, corresponds to the sum of all forces exerted by a set of other agents:

$$F_i^d(t) = \sum_{j,\, j\neq i}^{s} \varphi_j F_{ij}^{\ d}(t) = G(t) \sum_{j,\, j\neq i}^{s} \varphi_j(0,1) \frac{M_i(t) M_j(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \tag{5}$$

with *s* representing the agents swarm size. Each of the forces exerted in agent *i* by every other agent *j*, is randomly disturbed by using a uniformly generated random factor φ, generated in the interval [0,1]. The law of motion allows to evaluate the acceleration of agent *i*, for dimension *d* as:

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} = G(t) \sum_{j,\, j\neq i}^{s} \varphi_j(0,1) \frac{M_j(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \tag{6}$$

with M_j evaluated using the following expression establishing the relative importance of each individual mass, m_j, relative to the sum of all individual masses:

$$M_j = \frac{m_j(t)}{\sum_{j=1}^{s} m_j(t)}, \qquad m_j(t) = \frac{fit_j(t) - worst(t)}{best(t) - worst(t)} \tag{7}$$

with: fit_j representing the current iteration fitness value for agent *j*, *best*, *worst* represent the population best and worst fitness values in iteration *t*, respectively. It is relevant to note that individual mass, m_j, is a real number in the interval [0,1]; Moreover: i) when agent *j* is the best agent**,** $m_j(t)$=1; ii) when agent *j* is the worth agent, $m_j(t)$=0. In every iteration *t*, the agents velocity *v*, and position *x*, are changed respectively by:

$$v_i^d(t+1) = \varphi_i\, v_i^d(t) + a_i^d(t), \qquad x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{8}$$

The basic GSA steps are presented in Algorithm 1.

```
t = 0
initialize swarm X(t),
while(!(termination criterion))
   evaluate X(t)
   update G, best, worst
   for i=1 to s
        evaluate agents M_i and a_i
        update agents velocity, v_i
        update agents position, x_i
   end for
   t=t+1
end while
```

Algorithm 1: GSA basic algorithm.

As it can be observed, from expression (5) every agent exerts forces in all the other agents, often termed fully connected (or fully informed) model. However, the original algorithm considers also the possibility of using a set with the best agents. Considering this Elitism neighborhood model, the equation used to evaluate the acceleration (6) is adapted as follows:

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} = G(t) \sum_{j \in K_{best},\, j \neq i}^{s} \frac{M_j(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \tag{9}$$

The K_{best} size is gradually decreased from the full size swarm size to a minimum size along the search procedure. This elitism model is another way GSA can regulate exploration and exploitation.

As previously mentioned, some GSA hybrids incorporate PSO concepts. This is the case of the PSOGSA [9,10] and locally informed GSA (LIGSA) [13]. Here some concepts of the PSOGSA are presented. Differently than the PSO algorithm, the GSA does use the memorized global best agent position to guide the search. Indeed, the best global position can be lost from one iteration to the next in GSA, which may cause the algorithm premature convergence in a local optimum. Thus, the inclusion of a term with the global best agent position was hybridized into the GSA, by modifying the velocity equation as follows [13]:

$$v_i^d(t+1) = \varphi_i\, v_i^d(t) + c_1 a_i^d(t) + c_2 (g^d(t) - x_i{}^d(t)) \tag{10}$$

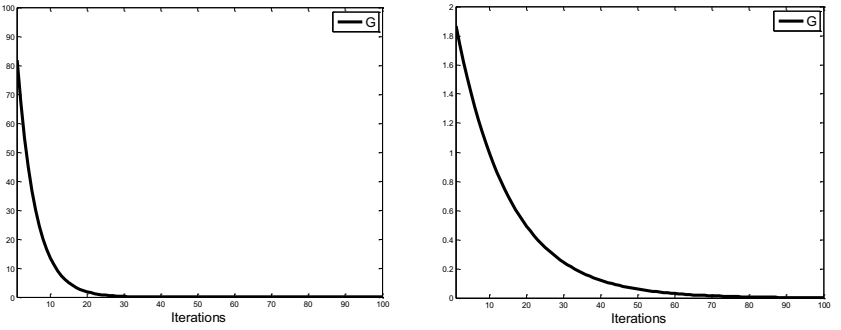
$$v_i^d(t+1) = \varphi_i\, v_i^d(t) + c_1 G(t) \sum_{j \in K_{best},\, j \neq i}^{s} \frac{M_j(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) + c_2 (g^d(t) - x_i{}^d(t)) \tag{11}$$

with K_{best} replaced in [13] by a local neighborhood, K_{local} for each agent i. Here instead of using a local neighborhood, classical GSA models are used. Constants c_1 and c_2, which in the PSO algorithm are known as the cognitive and social constants were proposed in [13] to be evaluated as follows:

$$c_1 = 1 - \frac{t^3}{t_{\max}}, \quad c_2 = \frac{t^3}{t_{\max}} \tag{12}$$

An issue overlooked in most GSA works is the parameter settings used to generate the gravitational constant. Equation (4) depends on three parameters G_0, α and t_{max}. Considering the original settings proposed in [1] by using G_0=100, α=20 and assuming a small number of iterations, e.g. t_{max}=100, outcomes in the G profile presented in Figure 1a). As it can be observed the initial value for this setting is $G(1) \approx 81.9$, and the decaying rate is quite fast which makes the G values with values near zero from iteration 40 until 100. This may cause premature convergence and stagnate the search in an early stage. As the gravitational constant is a multiplication factor in the agent's acceleration formula (9), depending on specific problem fitness values, higher values

in the initial iterations may cause agents to frequently move out of the search boundaries. Thus, it is convenient to select an appropriate G_0 and α for each specific case. Figure 1b), presents a G profile using the following settings: G_0=2, α=7, which has a smoother decaying ratio and lower $G(1)$.



a) With G_0=100, α=20 and t_{max}=100. *b) With G_0=2, α=7 and t_{max}=100.*

Fig. 1. Gravitational constant decaying profiles using (4).

Other functions can be used to generate the gravitational constant. An example is the linear decaying function represented by (13). By using (13) the search should present a much more unsteady behavior in the final search stage than using (4), due to the larger values for G in a latter evolution stage.

$$G(t) = G_0\left(1 - \frac{t}{t_{\max}}\right) \tag{13}$$

An alternative formula is presented in [12] which allows to decrease the parameter α in (4) along search as follows:

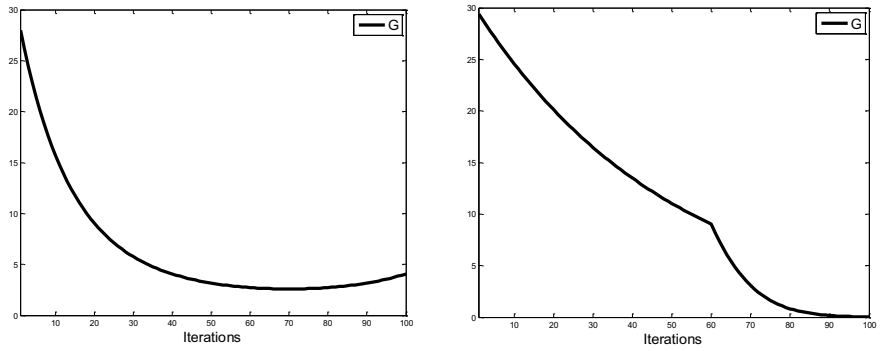
$$\alpha(t) = \alpha_{\max} - \frac{\alpha_{\max} - \alpha_{\min}}{t_{\max}} t \tag{14}$$

However, no information is provided on how to select the α maximum (α_{max}) and minimum (α_{min}) values, neither for G_0. Another study [17] proposes damping α in (4), using:

$$\alpha(t) = \begin{cases} \alpha_{\min} \Leftarrow \dfrac{t}{t_{\max}} < \sigma \\ \alpha_{\min} + \dfrac{\alpha_{\max} - \alpha_{\min}}{1 - \sigma}\left(\dfrac{t}{t_{\max}} - \sigma\right) \Leftarrow \dfrac{t}{t_{\max}} \geq \sigma \end{cases} \tag{15}$$

with $0 \leq \sigma \leq 1$. The interval $0.2 \leq \sigma \leq 0.8$ was found to be adequate to promote a good balance between exploration and exploitation. Examples of G profiles with G_0=30, α_{max}=7, α_{min}=2 using (14) and (15) with σ=0.6, are illustrated respectively in Figures

2a) and b). However, no rule is proposed to select α_{max}, α_{min} and G_0. Here a heuristic rule is proposed to select the initial gravitational constant in value in (4) approximately equal to half of the search interval for each dimension,



a) Using (14) with G_0=30, α_{max}=7, α_{min}=2 and t_{max}=100.

b) Using (15) with G_0=2, α=7, σ=0.6 and t_{max}=100.

Fig. 2. Gravitational profiles using (14) and (15).

Another relevant issue incorporated in the initial GSA algorithm proposal and freely available computational implementation [16], is the way that agents generated outside the search space boundaries can either be reinitialized within the search space limits or be clamped to the limit values. When the search space limits are known a priori, which is the case when using benchmark functions, the former can be more effective. However, in many practical problems defining the search space limits can also be part of the problem, as there is no clue where the optimal solution may be. In these cases, the convergence of a variable to an interval limit may indicate the necessity to widen the search space, making the initial gravitational constant value appropriate selection yet more relevant.

3 Simulation Results and Discussion

Due to space limitations just the well-known Ackley function is considered here:

$$f_1(x_i) = -a\exp\left(-b\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(c\,x_i)\right) + a + \exp(1), \qquad (16)$$
$$a = 20,\ b = 0.2, c = 2\pi,\quad x_i \in [-32.768, 32.768]$$

considering a two-dimensional space (d=2). Consider minimizing f_1 (16) with GSA and PSOGSA. The test conditions are the following: population size s=5, termination criterion t_{max}=50, randomly re-initializing particles generated out of the search interval. Two sets of conditions were used to generate the gravitational constant G using expression (4): *i)* with the originally proposed settings G_0=100, α=20 ii) with G_0=30,

α=7. In the case *ii)* G_0 was set using a value near half of the search space amplitude per dimension. The PSO was used considering the same number of objective function evaluations per algorithm run and ω linearly decreased in the interval [0.9, 0.4]. The results obtained for 100 runs per algorithm are presented in Table 1.

Table 1. Test results for function (16) with GSA, PSOGSA and PSO

Algorithm	Successful runs
GSA (G_0=100, α=20)	66
PSOGSA (G_0=100, α=20)	66
GSA (G_0=30, α=7)	96
PSOGSA (G_0=30, α=7)	96
PSO	99
GSA with K_{best} (G_0=30, α=7)	100

As it can be observed from the results in Table 1, for this function, there is no difference between the success rates obtained with standard GSA and PSOSGA. However, the results obtained with specific adjusted settings for equation (4) G_0=30, α=7, are significantly better than the ones obtained with G_0=100, α=20, thus providing strong indications that these parameters should be adjusted case by case. The PSO achieved a slightly higher success rate than GSA and PSOGSA in these test conditions. However, if a K_{best} elite model is used in the GSA, starting with the population size and gradually decreased with time, a success rate of 100% is achieved.

4 Conclusion

The gravitational search algorithm was reviewed, and insights about key issues concerning this metaheuristic presented using a tutorial approach. Topics regarding the selection of the initial gravitational search constant and forms to decay it along the search were addressed with more emphasis. Moreover, a heuristic rule is proposed to select the initial gravitational constant value and a hybrid approach among gravitational search algorithm and particle swarm optimization was tested along the original standard non-hybridized metaheuristics. Preliminary simulation test results were presented for a simple continuous function optimization example. The results analysis presented provided strong indications regarding the necessity to adjust the gravitational constant heuristic parameter accordingly to the problem specificity. Further work is necessary to consolidate the experimentation, also by considering other neighborhood elitism settings.

Acknowledgments. This work is funded (or part-funded) by the ERDF – European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project «FCOMP - 01-0124-FEDER-022701».

5 References

1. Rashedi E., Nezamabadi-pour H., and Saryazdi S.: GSA: A Gravitational Search Algorithm. Information Sciences, 179: 2232-2248, (2009).
2. Precup R.-E., David R.-C., Petriu E., Preitl S. and Rădac M.-B.: Gravitational Search Algorithms in Fuzzy Control Systems Tuning. Preprints of the 18th IFAC World Congress, pp. 13624-13629, (2011).
3. Rashedi E. and Nezamabadi-pour H.: A stochastic gravitational approach to feature based color image segmentation. Eng. Applic. of Artificial Intelligence. 26: 1322–1332, (2013).
4. Chen Z., Xiaohui Y., Tian H. and Ji B.: Improved gravitational search algorithm for parameter identification of water turbine regulation system. Energy Conversion and Management 78: 306–315, (2014).
5. Xiang J., Han X. H., Duan F., Qiang Y., Xiong X. Y., Lan Y. and Chai H.: A novel hybrid system for feature selection based on an improved gravitational search algorithm and k-NN method. Applied Soft Computing 31: 293–307, (2015).
6. Oliveira P. B. M., Pires E. S. and Novais P.: Design of Posicast PID control systems using a gravitational search algorithm. Neurocomputing 167:18–23, (2015).
7. Sarjila R., Ravi K., Edward J., Kumar K. and Prasad A.: Parameter Extraction of Solar Photovoltaic Modules Using Gravitational Search Algorithm. Journal of Electrical and Computer Engineering Volume (2016), Article ID 2143572, 6 pages http://dx.doi.org/10.1155/2016/2143572.
8. Ghavidel S., Aghaei J., Muttaqi K. and Heidari A.: Renewable energy management in a remote area using modified gravitational search algorithm. Energy 97: 391-399, (2016).
9. Mirjalili S. and Hashim S.: A New Hybrid PSOGSA Algorithm for Function Optimization. Proc. of the Int. Conf. on Computer and Information Application, pp. 374-377, (2010).
10. Mirjalili S., Hashim S. and Sardroudi S.: Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. Applied Mathematics and Computation 218:11125–11137, (2012).
11. Jianga S., Wanga Y. and Jiaa Z.: Convergence analysis and performance of an improved gravitational search algorithm. Applied Soft Computing 24:363–384, (2014).
12. Das P., Behera H. and Panigrahi B.:A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning. Swarm and Evolutionary Computation 28:14–28, (2016).
13. Sun G., Zhang A., Wang Z., Yao Y. and Ma J.: Locally informed gravitational search algorithm. Knowledge-Based Systems 104:134–144, (2016).
14. Suna G., Zhanga A., Yao Y. and Wang Z.: A novel hybrid algorithm of gravitational search algorithm with genetic algorithm for multi-level thresholding. Applied Soft Computing 46:703–730, (2016).
15. Gauci M., Dodd T. J. and Groß R: Why 'GSA: a gravitational search algorithm' is not genuinely based on the law of gravity. Natural Computing, 11:719–720, (2012).
16. Darzi S., Tiong S., Islam M., Soleymanpour H. and Kibria S.: An experience oriented-convergence improved gravitational search algorithm for minimum variance distortion less response beamforming optimum. PLoSONE11,doi:10.1371/journal.pone.0156749, (2016).
17. Rashedi R.: GSA source code. https://www.mathworks.com /matlabcentral/ fileexchange/ 27756-gravitational-search-algorithm--gsa-, Retrieved in 22-3-2017.

The Evolutionary Deep Learning based on Deep Convolutional Neural Network for the Anime Storyboard Recognition

Saya Fujino[1], Taichi Hatanaka[1] Naoki Mori[2], and Keinosuke Matsumoto[2]

[1] College of Engineering Osaka Prefecture University, 1-1 Gakuencho, Nakaku, Sakai Osaka 599-8531, Japan,
fujino@ss.cs.osakafu-u.ac.jp
[2] Graduate School of Engineering Osaka Prefecture University, 1-1 Gakuencho, Nakaku, Sakai Osaka 599-8531, Japan

Abstract. Recently, the researches of image recognition have been developed remarkably by means of the deep learning. In this study, we focused on the anime storyboards and applied deep convolutional neural networks (DCNNs) to those data. There exists one problem that it takes a lot of effort to tune DCNN hyperparameters. To solve this problem, we propose a novel method called evolutionary the deep learning (evoDL) by means of genetic algorithms (GAs). The effectiveness of evoDL is confirmed by computer simulations taking a real anime storyboard recognition problem as an example.

Keywords: anime storyboard, deep convolutional network, genetic algorithm, hyperparameter

1 Introduction

In the machine learning fields such as the image recognition, the deep learning technique has been one of the key issues because of its remarkable high performance. One of the most powerful approaches to solve image recognition problems is the deep convolutional neural network (DCNN)[1].

In this paper, we focus on the image of anime storyboard. The representation of an anime storyboard is quite rough compare to general comics but contains important information to help making final anime products.

In this paper, we propose a new hyperparameter tuning method for DCNN by means of an evolutionary approach to reduce parameter tuning cost. We adopt the Genetic Algorithm (GA)[2], which is a search and optimization algorithm based on the mechanism of the natural evolution as an evolutionary computation. The proposed method called "Evolutionary Deep Learning" (evoDL) is expected to find novel architectures of the deep learning. In this paper, we only focus on the tuning of numerical hyperparameters. To show the effectiveness of evoDL, computer simulations are carried out taking a real anime storyboard image as an example.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_34

2 Related works

A storyboard is one of the popular human sketches with text. In a past research[3], large-scale exploration of human sketches has been performed.

P. Bertla et al. also proposed recognition of sketches[4] by a multi-class SVM classifier utilizing a binary decision tree.

From the view point of neuroevolution, the HyperNEAT which adopts indirect encodings for network has been proposed[5]. Comparison study between neuroevolution and SGD based methods has been reported[6]. The research of combining the GA with DCNN is reported[7]. Although several other researches have been proposed[8], there are no researches applying DCNN to an anime storyboard directly.

3 Storyboard of an Anime

In this section, we explain the outline of an anime storyboard.

3.1 Storyboard

A storyboard is the first step of creating animation. It's a graphic organizer in the form of illustrations or images displayed in sequence for the purpose of shooting a motion picture, anime and motion graphic. This is expressed by using not only illustrations but simple text which explains the situation for target cut.

In an anime, the storyboard is one of the most important parts because the quality of a final anime is strongly dependent on pre-visualizing a motion picture based on storyboards. It is known that creating a final anime from storyboard takes much effort.

3.2 Example of a Anime Storyboard

Figure 1 shows the example of real Japanese anime storyboards. We regard anime storyboards as intermediate products products between anime and comics.

4 Deep Convolutional Neural Network

A DCNN is one of the deep learning methods and is widely used in the image recognition field. In this paper, we used the architecture of AlexNet[1] as DCNN. AlexNet is one of the most popular architectures of a DCNN. It contains eight learned layers: the first five are convolutional and the remaining three are fully connected. Using a softmax function, the last fully connected layer produces an output consisting of class labels.

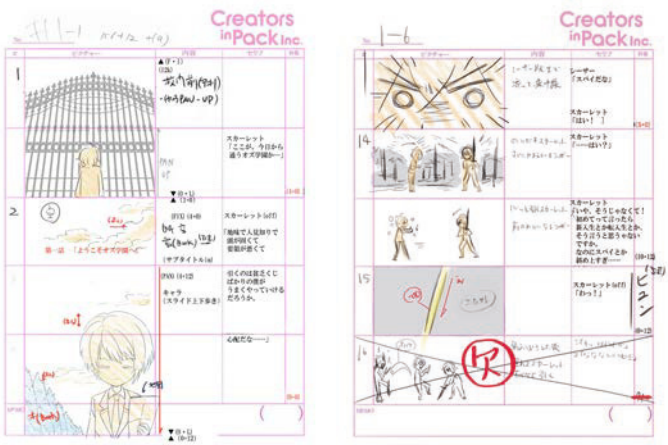


Fig. 1. Example of anime storyboards (©Production committee of Poni-Pachet SY/HOBIBOX/OZMAFIA!![9])

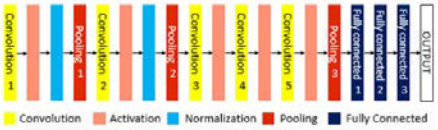


Fig. 2. The architecture of AlexNet

5 Evolutionary Approach for DCNNs

We propose a novel evolutionary approach for finding a good deep learning model called "Evolutionary Deep Learning" (evoDL). We have adopted the genetic algorithm (GA) as the optimization method fo evoDL.

In this study, we only focus on hyperparameters represented by an integer or a boolean value such as the number of filters, the filter size and the max pooling size, using an activation function. We used the conventional GA framework except for the representation of the genotype. The uniform crossover, the uniformly distributed mutation and the tournament selection are applied in this respective order to the population of the genetic algorithm.

5.1 Representation

Table 1 shows the details of alleles of the genotype in GA.

Figure 3 shows the example of the chromosome of an individual. Each string in the locus is related to strings in table 1. The basic structure of DCNN is fixed

Table 1. The gene of individuals in GA

Design Variables	Allele
The number of filter (NF)	8, 16, 32
Filter size (FS)	3, 5, 7
Pooling size (PS)	3, 5, 7
NL1	64,128, 256, 512
NL2	16, 32, 64, 128
Batch size (BS)	5, 10, 15
Activation function with ReLU (Re)	1 (use), 0 (not use)

Nl1: The number of node in fully connected layer 1
Nl2: The number of node in fully connected layer 2

Fig. 3. An example of genotype

according to AlexNet. In figure 3, the seven genes "Re" represent the flag of ReLU functions which are set just after three convolutional layers and two fully connected layers. If gene Re is 1, the corresponding ReLU function is set after the target layer.

5.2 Fitness Function

In this study, the fitness function $F(s)$ of evoDL is the negative value of sum of loss in k-fold cross validation using training data, where s is an individual related to certain DCNN.

This optimization problem is represented as follows:

$$\arg\max_{s} F(s) = -\sum_{i=1}^{k} f_{\text{loss}_i}(s) \tag{1}$$

where s denotes an individual and $f_{\text{loss}_i}(s)$ is the loss of k-fold cross validation after i epochs calculated by cross entropy. In this case, the maximum fitness value is 0. Another kinds of fitness functions such as sum of accuracy or function of considering complexity of network are available in proposed evoDL.

6 Experiments

Here, we show the experimental results of evolving a DCNN model by GA. The main purpose of this experiment is to confirm the validity of proposed evolutionary method. We also investigate the DCNN ability of recognizing an anime storyboard.

Table 2. Computer spec of experiments

OS	Ubuntu 14.04.4 LTS
CPU	Intel(R) Core(TM) i7-5930K CPU @ 3.50GHz
Memory	32.0 GB
GPU	NVIDIA GeForce GTX TITAN X (12 GB)

6.1 Target Problem

Since lots of anime storyboards are drawn with several colors such as black, red and/or blue, we utilized RGB anime storyboard images as the recognition problem. We utilized two animes which were broadcast in 2014 $\sim$ 2015 in Japan. Both anime contain 12 stories, and each storyboard is composed of about 100 pictures like shown in figure 1. We denote two animes as Anime1 and Anime2[3].

In this experiment, the DCNN model is evolved to recognize two classes of the RGB anime storyboards images with size 100×141. Input channel of DCNN is 3. In this study, the training data are storyboards data of 5 stories (episode 1 to 5), and test data are test data are remaining ones (episode 6 to 12). Anime1 is set to class 1, and Anime2 is set to class 2.

Training data in evolution The number of storyboard data of class 1 is 860, and the one of class 2 is 815. The total amount of training data is equal to 1675. In GA parts of evoDL, the fitness in 5.2 is calculated by using the results of a three-fold cross validation. Each individual is translated into a DCNN model and trained.

After this, we obtained this DCNN accuracy and loss. Because of the three-fold cross validation, we repeated this 3 times and obtained the fitness value using eq. (1).

Test data In test data, we focus on each episode because the storyboard is strongly dependent on certain episode. We would like to check robustness of DCNN obtained by evoDL.

6.2 Experimental Conditions

Table 2 shows computer spec of this experiments. We used a flexible framework for the deep learning library chainer[10] in a DCNN part, and Java for GA part in evoDL. Table 3 shows the setting of GA. In addition, we set the maximum number of generation to 12 and population size to 16. Although we understand those numbers seem to be not enough to achieve effective evolution, we use those settings as first step of an evolutionary approach to obtain DCNNs because evoDL requires a large amount of computational resources.

In chainer, we used Adam as optimizer. The whole process of evoDL evolution is as follows:

[3] Because of copyright, we cannot show the real title.

Table 3. Setting of GA in evoDL

generation size	12
population size	16
chromosome length	22
crossover type	uniform
crossover rate	1.0
mutation rate of each locus	$\frac{1}{L}$ (L is chromosome length)
selection	tournament selection
tournament size	3
Elitism	true

1. DCNN models are evolved in GA by using training data with three-fold cross validation. To make the number of total taring data equal, batch size and the number of epochs (batch, epoch) are set to (5, 150), (10, 75) or (15, 50) in estimating GA individuals.
2. We obtained best hyperparameters of DCNN model from an elite individual of GA in the final generation.
3. DCNN models are trained with obtained hyperparameters. In this step, max number of epoch is 2000 and all training data are used because cross validation is unnecessary. .

6.3 Results and Discussion

In this experiment, because the dataset is not so large, our system can find a fine individual for the DCNN with ease. The recognition task of present dataset may be too easy for our proposed method. Applying our system to a more difficult dataset is one of the most important further studies.

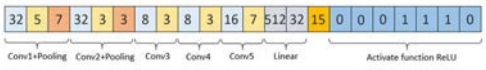


Fig. 4. The genotype of best individuals in test data

Figure 4 shows the elite individuals genotype which performs best with a train accuracy of 1.0.

Table 4 shows the results of applying best DCNN model with hyperparameters shown in figure 4 after 500 and 2000 epochs of training. In this table, columns of Anime1 and Anime2 show the creator represented by a capital letter and the number of image data in storyboard is shown in parentheses. Since the production staff of Anime1 and Anime2 are overlapped, several storyboard creators are appeared in both Anime1 and Anime2. We tried the following two experiments:

E1 Episodes 1 to 5 are used as training data, and Episodes 6 to 12 are used as test data.

E2 Shuffle all 12 storied data and divide those by the ratio 5:7. 5/12 data are used for training data and 7/12 data are used for test data.

The 2 bottom lines of table 4 show the weighted average accuracy of **E1** and the accuracy of using shuffled data in **E2**.

From comparison results of **E1** between 500 epochs and 2000 epochs, accuracy rates of 2000 epochs are higher than that of 500 epochs in all episodes except 6 and 11. However, the differences are only $1 \sim 2\%$ in episodes 6 and 11. On the other hand, accuracy rates of 2000 epochs in episodes 9 and 12 is almost 10% higher than that of 500 epochs. Therefore, we focus on only results of 2000 epochs in the following.

In 2000 epochs results of **E1**, accuracy rates of episode 6, 7 are very high (over 98%). Accuracy rates of episodes 9, 11 and 12 also high (over the 80%). On the other hand, the accuracy rates of episode 8 and 10 are even lower than baseline 0.513 ($\approx 860/(860 + 815)$). The reason for those low accuracy rates are the creator "A". "A" appears twice in episodes 1 and 5 of Anime1, while "A" handles storyboard of Anime2 in both episodes 8 and 10. Although "A" also appears in the episode 3 of Anime2, "A" may have a stronger influence on Anime1 than on Anime2 because "A" is the director of Anime1. That is why, the differences in accuracy rates among episodes are large, and the weighted average becomes under 80%.

From results of **E2**, DCNN shows high accuracy rates both 500 and 2000 epochs; 96% and 97%. Those results show that experiments using shuffled data are easier than the one using story based data. This is because training data of **E2** contains all creator's images because of shuffling. Not only anime title but the numbers of episode and creator are important to analyze anime storyboards. Those results also show that it is possible to be able to learn the style of drawing of creators by DCNN.

7 Conclusion

In this study, we proposed a novel and evolutionary approach for the deep learning called evoDL and applying evoDL to obtain the DCNN model for the recognition of the anime storyboards. Using AlexNet as the basic DCNN model, we obtained a DCNN model for tuning hyperparameters by the evoDL. Extend evoDL to be able to represent real number hyperparameters and complex network topology is the important future works.

A part of this work was supported by JSPS KAKENHI Grant, Grant-in-Aid for Scientific Research(C), 26330282. A part of this work was also supported by JSPS KAKENHI Grant, Grant-in-Aid for JSPS Fellows, 16J10941.

Table 4. Results of applying a CNN to test data

	Storyboard Creator		Accuracy	
	Anime1	Anime2	500 epoch	2000 epoch
episode 1	A (215)	C , K (198)	train	train
episode 2	B (155)	C , G (149)	train	train
episode 3	D (135)	A (162)	train	train
episode 4	C , B (133)	L , K (217)	train	train
episode 5	A (177)	M , B (134)	train	train
episode 6	E (114)	N , B (115)	0.99	0.98
episode 7	D (173)	O (139)	0.99	1.0
episode 8	F (161)	A (149)	0.47	0.49
episode 9	B (242)	P (169)	0.83	0.92
episode 10	G (158))	A (185)	0.43	0.48
episode 11	H , I (226)	B (170)	0.90	0.88
episode 12	J (153)	Q (127)	0.73	0.83
weighted average accuracy(**E1**)	-	-	0.76	0.79
shuffled data accuracy(**E2**)	-	-	0.96	0.97

References

1. A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In P. Bartlett, F.c.n. Pereira, C.j.c. Burges, L. Bottou, and K.q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pp. 1106–1114. 2012.
2. D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989.
3. M. Eitz, J. Hays, and M. Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, Vol. 31, No. 4, pp. 44:1–44:10, 2012.
4. K. Matsumoto P. Bertola, N. Mori. Sketch recognition for interactive multi-agent system. *Proc. of the 58th Annual Conference of the Institute of Systems, Control and Information Engineers (SCI'14)*, No. 334–4, 2014.
5. J. Gauci and K. O. Stanley. Autonomous evolution of topographic regularities in artificial neural networks. *Neural Computation*, Vol. 22, No. 7, pp. 1860–1898, 2010.
6. G. Morse and K. O. Stanley. Simple evolutionary optimization can rival stochastic gradient descent in neural networks. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference*, pp. 477–484. ACM, 2016.
7. P. Yunming Y. Zhining. The genetic convolutional neural network model based on random sample. *International Journal of u- and e- Service, Science and Technology*, Vol. 8, No. 11, pp. 317–326, 2015.
8. S. Ding, H. Li, C. Su, J. Yu, and F. Jin. Evolutionary artificial neural networks: a review. *Artificial Intelligence Review*, Vol. 39, No. 3, pp. 251–260, 2013.
9. Creators in pack. `http://anime.ozmafia.com/`.
10. chainer. `http://chainer.org/`.

Preliminary Study of Multi-modal Dialogue System for Personal Robot with IoTs

Shintaro Yamasaki and Kenji Matsui

Department of Robotics,
Osaka Institute of Technology,
1-45 Chayamachi, Kita-ku, Osaka, 530-0013 Japan
{m1m16h20,kenji.matsui}@oit.ac.jp

Abstract. Personal robot is a new technological solution where various IoT machines collaborate and provide new type of services. In this research, we develop a cloud-network based personal robot platform technologies, which offer advanced IoT services through its cost-effective strategy. As for the man-machine interaction, two stage dialog system was implemented, i.e. local side and the cloud server side. Also the cloud server has an ability to download various IoT applications. In this study, we tested the proposed system using smartphone as one of the IoT devices using data retrieval application. Our preliminary usability test showed good overall performance, however, we found that the interaction timing needs to be improved.

Keywords: dialogue system, personal robot, cloud, mobile, IoT

1 Introduction

A personal robot is one which interacts with people mainly at home and provides services for individuals. The market size of personal robots (virtual & physical) have been rapidly growing [1] and possibly continue to grow at home as well as other IoT devices in the future [2].

Personal robot technologies are promising and commercially available now, but not widely utilized yet. There are mainly two issues. The one issue is that often those robots are quite costly if you would like to have various functions such as display, sensors, actuators, etc. The other issue is it is difficult to keep providing services to meet peoples needs. Some of the robots, such as Pepper [3], Palro [4], Jibo [5], have features to be able to update the applications and even providing toolkits, however, such flexibility is often limited by available hardware resources. To solve those issues as stated above, we propose a new type of robot system that provides multiple services in cooperation with mobile devices, cloud servers and IoT devices, such as sensors, cars and home appliances.

The key strategy is to utilize around resources to maximize the functionality without adding own hardware. The cloud-network-based architecture enables downloadable application service system which users can browse and install apps

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_35

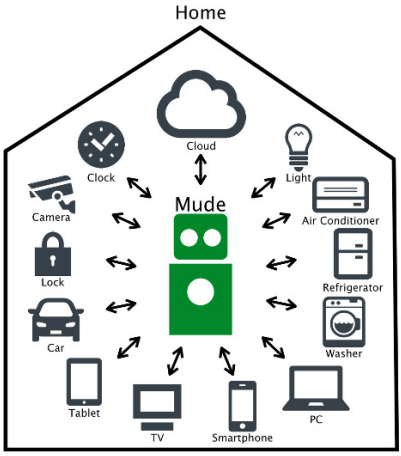


Fig. 1. Concept of the proposed Personal Robot system

in the robot system. Our experimental robot is called "Mude. Fig.1 shows the key concept of the proposed system.

In order to realize the proposed system, multimodal dialogue-based interaction is intuitive, natural and useful. One of our key features is to utilize not only the robot itself but also other IoT devices around the robot system. That concept is very important to make our system highly cost-effective. For example, using TV monitor, tablet-PC, or smartphone, the robot can utilize the display capability even the robot itself does not have any display function. In that sense, with minimum amount of resources, the robot can provide various functionality borrowing from other IoT devices. As for the man-machine interaction, two stage speech dialog system was implemented, i.e. local side and the cloud server side. Also the cloud server has an ability to download various robot applications. Tamagawa A. [6] has evaluated the robot agent system, which the pet robot "Phyno" guides rooms with showing a map on the display of mobile device, and obtained high score of evaluation in terms of convenience. In this study, we tested the proposed system using smartphone as one of the IoT devices. The system cooperates with existing smartphone applications to provide detailed information to the users and reduce the cost for resources of the robot and development of smartphone application for additional robot application. This paper describes the initial prototype system development and the results of basic evaluation.

2 Multi-modal dialogue system

2.1 Architecture

Fig.2 shows the architecture of the multimodal dialogue system which we are considering. This system is separated into two parts, local side (front-end) and

cloud server side (back-end). And the system structure of the initial prototype is shown Fig.3.

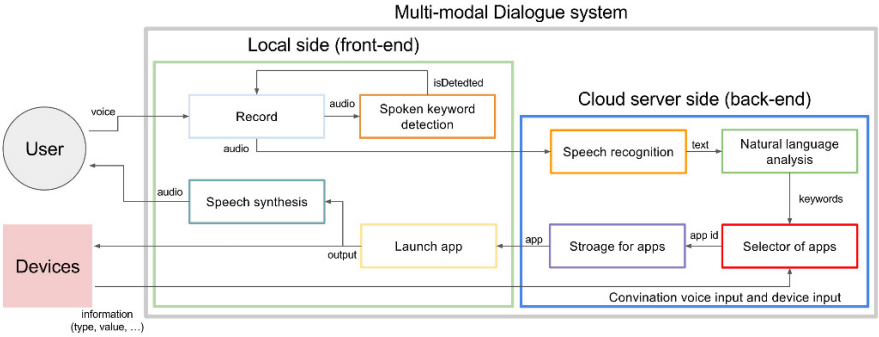


Fig. 2. Architecture of our dialogue system

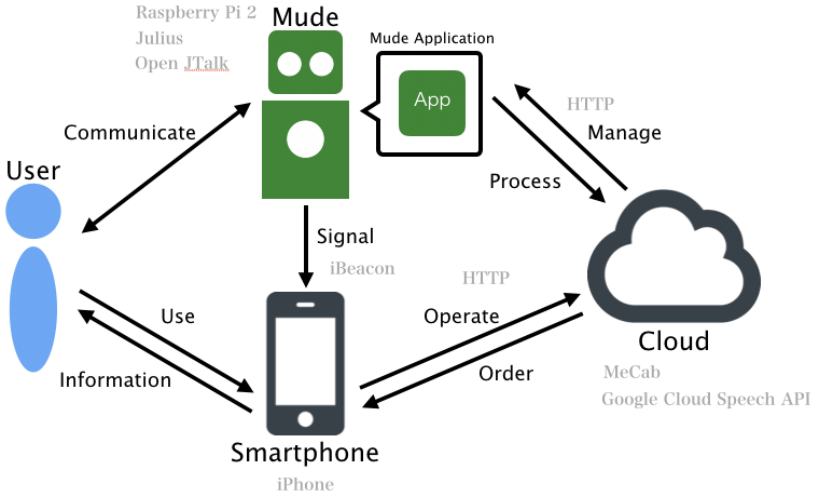


Fig. 3. Structure of initial prototype

The local side handles the inputs, which are speech input from users and information from IoT devices, and the outputs from applications, which are selected at cloud server side depending on the input data and provide services to users. It mainly has four elements as shown below:

Record is to record users speech from a microphone.

Spoken keyword detection detects the keyword which is a trigger to invoke dialogue between the user and the system. This contains simple speech recognition system to assess if the user says a keyword or not. For initial prototype, Julius [7] is used.

Launch app launches an application selected by a selector at the cloud server side depending on the inputs. It outputs the services.
Speech synthesis is the text to speech function to output the response of users input as audio from a speaker. Initial prototype uses OpenJTalk [8].

These parts are implemented onto personal robot.

Otherwise, the cloud server side handles the processes which take a lot of resources to calculate. It mainly has four elements as shown below.

Speech recognition converts speech to text. This has larger vocabulary than "Spoken keyword detection" on local side. For initial prototype, Google Cloud Speech API [9] is used.
Natural language analysis analyzes the text as the result of speech recognition. Initial prototype uses MeCab [10] for morphological analysis.
Selector of apps selects application which provides services to users depended on the result of natural language analysis and information of IoT devices.
Storage of apps contains applications.

In summary, this cloud-network based personal robot system is consist of those two parts as shown above to handle all processes smoothly and with no significant delay even large processes are moved on to cloud server side which has higher performance.

2.2 Spoken dialogue

This multimodal dialogue system has two stage structured for the man-machine interaction mechanism. One is on local side, the other is on cloud server side. The local side one is the trigger to begin conversation with users once it detects keywords in users speech. The speech recognition system has small vocabulary and requires no significant computational power so that it can be installed into relatively simple hardware. On the other hand, the main speech recognition function is on the cloud server side which has high computational power resources.

2.3 Application system

We designed the application software architecture for this dialogue system and developed a prototype of downloader, management system, and downloading capability for the local side by using smartphone also for the cloud side.

These applications are stored into the storage on the cloud side with some information for the selector described below.

Application selector Application selector is to select application which provides services depending on user's input and information of IoT devices. It is implemented on the cloud server side.

Some examples of the possible applications are: by communicating with clowd server, playing music, data retrieval, weather information, and checking schedule,

also by collaborating with IoT devices, showing maps/photos, notifying alert and controlling home appliances.

This selector estimates suitable application based on the following Keywords and parameters in Actions registered with application file onto database at the cloud server.

Keywords are words which refer to the function and the service which the application has.
Actions contain action commands and parameters.

The requested application is browsed and selected based on the extracted keywords from the spontaneous input speech. Then, the selected application will be executed once all of the required action parameters have gathered.

2.4 Man-machine interaction

The local side of this dialogue system has a multimodal user interface which is not only spoken dialogue but also touch screen, display, and sound capability using mobile phone resources.

The initial prototype, used smartphone (Fig.4) so that users can receive more detailed I/O, such as pictures, documents, and maps, than only speech.

Fig. 4. Initial prototype and smart phone

3 Usability Evaluation

Eleven subjects evaluated the usability of Mudes multi-modal user interface in the case of utilizing smartphones. They tested three tasks as shown below after they learned the system overview and the usage.

1. Want to know where Osaka station is. (by Map application)
2. Want to eat Pasta and look for a good restraint. (by Restraint application)

3. Want to know tomorrow's schedule. (by Schedule application)

After the test, they were asked to answer the fifteen questions in terms of the user interface performance. They rated each question using seven scales, then the performance was evaluated by SD method [11] [12]

As the result, the prototype system got a good overall feedback. However, Naturalness, Comfortableness, Usefulness were evaluated poorly. One of the possible reason is that it is difficult to know when to speak, and another issue is the delayed response. (Fig.5).

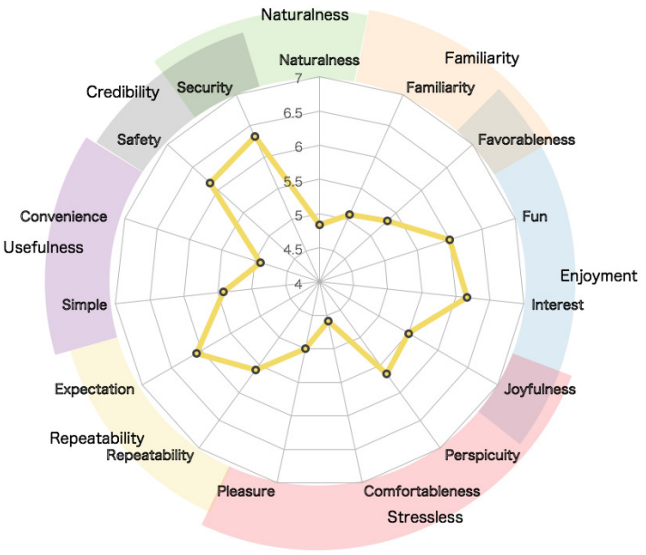


Fig. 5. The result of usability evaluation

4 Conclusions

This paper described the cloud-network based personal robot platform technologies, which offer advanced IoT services through its cost-effective strategy. We developed two stage dialog system using local side and the cloud server side in order to make speech recognition stable and accurate. Also we made the cloud side as a server of various IoT applications. We tested the proposed system using smartphone as one of the IoT devices using data retrieval applications. Our preliminary usability test showed good overall performance except some of the interaction timing needs to be improved. We plan to make further performance evaluation using various applications.

References

1. Tractica.: "Nearly 100 Million Consumer Robots Will be Sold During the Next 5 Years". Tractica press release. Nov 23 (2015)
2. Ericsson.: "INTERNET OF THINGS OUTLOOK". Ericsson Mobility Report. Nov (2016)
3. Softbank Robotics.: "Pepper, the humanoid robot from Aldebaran, a genuine companion", https://www.ald.softbankrobotics.com/en/cool-robots/pepper
4. Fijisoft.:"Palro", https://palro.jp/en/
5. Jibo.: "Jibo Robot", https://www.jibo.com
6. Tamagawa, A., Tamamoto, K., Nakagawa, S.: Evaluation of Superiority of Robot Agent in Spoken Dialog Interface. IPSJ SIG Technical Report Vol.2013-SLP-97 No.11 (2013)
7. Julius.: "Julius: Open-Source Large Vocabulary Continuous Speech Recognition Engine", https://github.com/julius-speech/julius/
8. OpenJTalk.: "Open JTalk", http://open-jtalk.sourceforge.net
9. Google.: "CLOUD SPEECH API BETA Speech to text conversion powered by machine learning" https://cloud.google.com/speech/
10. MeCab.: "MeCab MeCab: Yet Another Part-of-Speech and Morphological Analyzer", http://taku910.github.io/mecab/
11. Yoshino, K., Mori, S., and Kawahara, T.: "Spoken Dialogue System Based on Information Extraction and Presentation Using Similarity of Predicate Argument Structures". IPSJ SIG Technical Report Vol. 52 No. 12 (2011)
12. Kanda, T., Ishiguro, H., Ishida, T.: "Psychological Evaluation on Interactions between People and Robot". Journal of the Robotics Society of Japan Vol. 19 No. 3 (2001)

Odor Classification for Human Breath Using Neural Networks

Sigeru Omatu

Osaka Institute of Technology, Osaka, 535-8585, Japan,
omtsgr@gmail.com,
https://www.oit.ac.jp

Abstract. The principle of metal-oxide gas sensors is based on oxidation and reduction rule. The aim of this paper is to classify odors of breath in which several gasses are mixed. Using neural networks to train a specific odor we classify mixed odors. First, we train a layered neural network for specific odor which is main component of our breath. Then we apply the breath data to classify the other components.

Keywords: odor measurement, breath classification, layered neural networks

1 Introduction

Sense of odor is one of five senses of human being. But compared with audiovisual senses, its study has not been paid attention since it is based on chemical reaction although it has been recognized for various applications in human life and industrial sectors. Based on the progress of medical research on the system of olphactory organs, artificial electronic nose (E-nose) systems have been developed. From technical and commercial viewpoints, various E-nose systems have been developed in the field of quality control of food industry [1], public safety [2], and space applications [3].

In this paper we propose a new E-nose system which can separate two kinds of odors of mixed molecuels by using a layered neural network. Historically, J. Milke [4] proved that two kinds of metal-oxide semiconductor gas sensor(MOGS) could have the ability to classify several sources of fire more precisely compared with a conventional smoke detector. However, his results achieved only 85% of correct classification by using a conventional statistical pattern classification.

An E-nose has been developed for odor classification of various sources of fire such as household burning materials, cooking odors, the leakage from the liquid petroleum gas (LPG) in [5],[6]. by using neural networks of layered type.

The purpose of this paper focuses on mixed odors classification. Various odors have been mixed naturally in our living environment. If a poisonous odor exists in our living environment, it must be recognized immediately by humans. Therefore, it is necessary to classify mixed odors into each component of mixed odors. However, sensors which classify odors of several types do not exist currently. The

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_36

research purpose is to classify various mixed odors and construct the system that can be used repeatedly and inexpensively. We attempt to measure mixed odors using commercially available sensors and classify mixed odors of various density using a neural network.

2 Odor Sensors and Sensing System

Gas sensors using a tin oxide was produced in 1968 [5],[6]. There are tin, iron oxide, and tungsten oxide as typical gas sensing substances which are used in metal oxide semiconductor gas sensors (MOG sensor). In this case, an electrical resistance of the tin becomes low. Using the above oxidation and reduction process, we could measure whether a gas appears or not.

In this paper, MOG sensors are used to measure the single odor and mixed odor. Table 1 shows the five sensors in the experiment.

Table 1. Gas sensors used in the experiment made by FIS Corporation.

Sensor number (Type)	Gas detection types	Applications
Sensor 1 (SB-AQ8)	volatile, organic compound	air quality
Sensor 2 (SB-15)	propane, butane	flammable gas
Sensor 3 (SB-42A)	freon	refrigerant gas
Sensor 4 (SB-31)	alcol, organic solvent	solvent
Sensor 5 (SB-30)	algohol	alcohol detection

3 Neuron Model and Neural Network

We show the neuron structure in Fig. 1. The left hand side denotes inputs $(x_1, x_2, x_3, dots, x_n)$, the right hand side is the output y_j where $x_i, i = 0, 1, \ldots, n$ denotes input data, $w_j k, k = 0, 1, \ldots, n$ show weighting coefficients, $f(\cdot)$ is output function of the neuron, and y shows output. Note that $x_0 = -1$ and $w_0 = \theta$ where θshows a threshold of the neuron. Although there are several types of the output function $f(\cdot)$, we adopt the sigmoid function.

Neural networks are classified in supervised learning and unsupervised learning. Among the supervised learning methods the most popular one is a layered neural network. It has input layer, hidden layer, and output layer as shown in Fig. 2.

4 Measurement of Odor Data

We have measured four types of odors as shown in Table 2. The sampling frequency is 500[ms], that is, the sampling frequency is 0.5[s]. The temperatures of

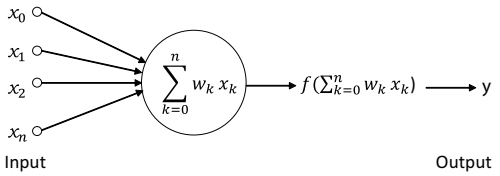


Fig. 1. Neuron model.

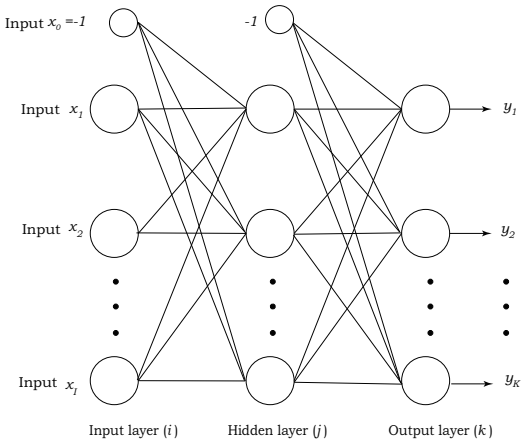


Fig. 2. Layered neural network.

odor gases were 18∼24[°C] and the humidities of gases are 20∼30[%]. We show sample paths of each odor of Table 1 in Figs. 3–6 where horizontal axis is time [s] and vertical axis is voltage [V]. Note that those factors in Table 2 are main odors included in our breath. Bad odor of our breath will make the neighboring persons unpleasant although the person himself might not notice it.

Furthermore, the bad odor of persons may reflect a kind of diseases and the measurement of human breath and odor classification will be profitable to predict a disease of persons in advance. Since these gasses exist very thin density of ppm or ppb order, we make those gasses of 1ppm∼5ppm order from 1% pure gasses. Using those gasses of densities, we have made mixing gasses such as 1:1, 1:5, and 5:1 mixing rates where a:b means a ppm and b ppm gasses are mixed. In case of four sources, there are six combination cases to mix two gasses. In this paper, we have mixed only two gasses of methyl-mercaptan (C) and hydrogen sulfide (D) since they are main components in our breath. Those sample paths are shown in Fig. 7–9 for different mixing rates of methyle-mercaptan (C): hydrogen sulfide (D) such as 1:1, 1:5, and 5:1, respectively.

Table 2. Training data.

Label	substance	samples
A	acet-aldehyde	3
B	ethylene	3
C	methyl-mercaptan	3
D	hydrogen sulfide	3

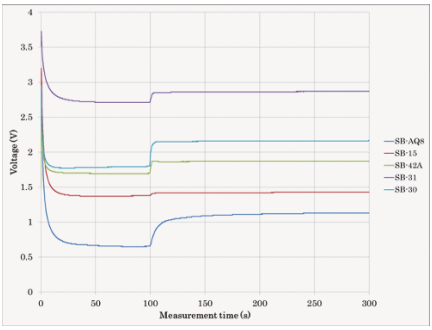


Fig. 3. An example of acet-aldehyde.

5 Odor Classification of Breath Data

We must extract the features of the sample paths. Those sample paths of Figs. 3-8 show that at t=0, dry airs blow and at 100 seconds later gasses are injected. We can see that after about 60 seconds since dry airs blew in and after about 60 seconds later since odor gasses were injected at t=100 seconds later all sensors would become almost steady. Therefore, we decided the changes from t=60 to t=160 for each sensor reflect the features of the gasses and take the difference values at t=60 seconds and t=160 seconds for each sensor as features.

6 Training for Classification of Odors

In order to classify the feature vector, we allocate the desired output for the input feature vector where it is five–dimensional vector as shown in Table 3 since we have added the coefficient of variation to the usual feature vector to reduce the variations for odors. The training has been performed until the total error becomes less than or equal to $.5 \times 10^{-2}$ where η=.2. Note that the training data set is the first sample data among three repeated data of A, B, C, and D. After that, the second sample data are selected as the training data set and the third samples data are selected as the training data set. The test data sets are selected as the remaining data sets except for the training data set.

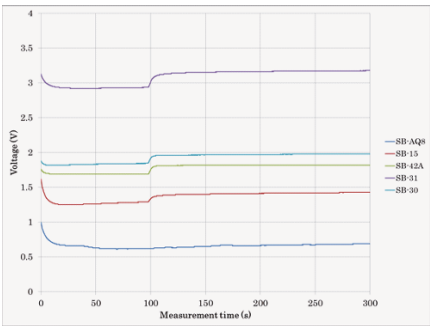


Fig. 4. Ethylene.

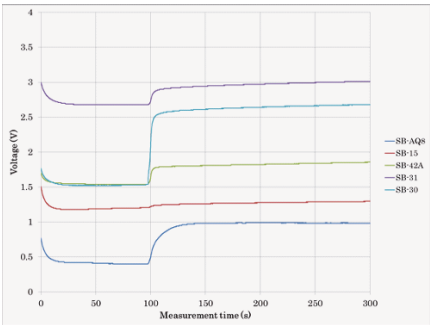


Fig. 5. Methyl-mercaptan.

7 Odor Classification Results

7.1 Single odor classification results

Using a layered neural network with five inputs and four output, we trained the neural network for one training data until the total error becomes 0.001. After training, we checked the remaining two data set. For two test data, correct classification rates became 100% and using the leave-one-out cross validation check, correct classification rates were100%.

7.2 Mixed odor classification results

Using the trained neural network, we tested the classification results whether the original two odors could be classified or not. The classification results are shown in Table 4. From this results, the classification results are not so good since there are many misclassification results came out. Especially, C&D has been classified acetaldehyde (A) because hydrogen sulfide (D) is too strong chemical reaction even if the very thin density levels as well as acetaldehyde (A). Thus,

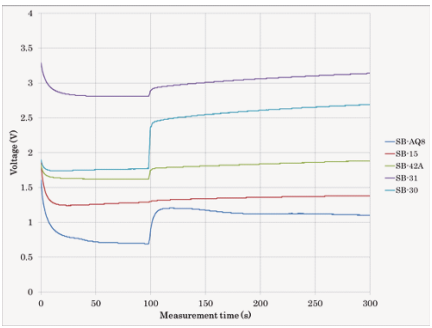


Fig. 6. Hydrogen sulfide.

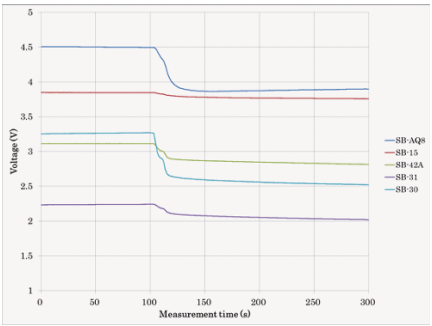


Fig. 7. Mixing rate 1:1.

we must reconsider our method to classify the mixed odors classification. One of the approaches is to reduce the maximum values of odor data in order to prevent a sole winner case.

Therefore, we use the following formula to reduce the effect of a sole winner.

$$z = x - \alpha y \tag{1}$$

where x is a feature, y is the top value of each row Table 4 and α is a constan which has been determined by using a genetic algorithm. In this case we have obtained α=0.3. Using this value, we have had the results of Table 5.

8 Conclusions

We have proposed a classification method of mixed odor sensing data using a layered neural networks. The mixed odors classification is not so simple as single odor classification cases. One way seems to use the mixed odors directly as mixed odor of two substances. In that case we need the classification boundary more

Table 3. Training data set for acetaldehyde (A), ethylene (B), methyl-mercaptan (C), and hydrogen sulfide (D).

Labels	A	B	C	D
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

Table 4. Classification results for mixed odors of methyl-mercaptan (C), and hydrogen sulfide (D).

Mixing rates	A	B	C	D
1:1 (C&D)	4	3	6	6
1:5 (C&D)	9	0	0	9
5:1 (C&D)	9	0	0	9

Table 5. Classification results when we reduce the largest odor data effect by (1).

Mixing rates	A	B	C	D
1:1 (C&D)	7	2	7	2
1:5 (C&D)	7	0	2	9
5:1 (C&D)	6	0	3	9

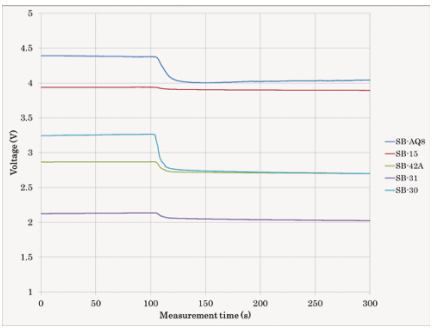


Fig. 8. Mixing rate 1:5.

and more big world. Therefore, we have proposed one way to classify the mixed odors classification from single classification neural network.

Acknowledgment

This research has been partially supported by JST and JKA foundations

References

1. Norman, A., Stam, F., Morrissey, A., Hirschfelder, M., and Enderlein, D. :Packaging effects of a novel explosion-proof gas sensor. Sensors and Actuator B. 114, 287–290 (2003)
2. Baric, N., Bucking, M., and Rapp, M. : A novel electronic nose based on minimized saw sensor arrays coupled with same enhanced headspace analysis and its use for rapid determination of volatile organic compounds in food quality monitoring. Sensors and Actuator B. 114, 482–488 (2006)
3. Young, R., Buttner, W., Linnel, B., and, Ramesham, R. ; Electronic nose for space program applications. Sensors and Actuator B. 93, 7-16 (2003)
4. Milke, J. :Application of neural networks for discreminating fire detectors, 1995 International Conference on Automatic Fire Detection, AUBE'95, Duisburg, Germany (1995)
5. Charumpom, B., Yoshioka, M., Fujinaka, T., and Omatu, S. :An e-nose system using artificial neural networks with an effective initial training data set. IEEJ. Trans. EIS, Japan, 123, 1638–1644 (2003)
6. Fujinaka, T., oshioka, M. Omatu, S. and Kosaka, T. : Intelligent electronic nose systems for fire detection systems based on neural networks, Int. J. of Adv. in Intelligent Systems. 2, 268–277 (2009)

Outline of a Generalization of Kinetic Theory to Study Opinion Dynamics

Stefania Monica and Federico Bergenti

Dipartimento di Scienze Matematiche, Fisiche e Informatiche
Università degli Studi di Parma
Parco Area delle Scienze 53/A, 43124 Parma, Italy
{stefania.monica,federico.bergenti}@unipr.it *

Abstract. This paper presents an analytic framework to study the dynamics of the opinion in multi-agent systems. In the proposed framework, each agent is associated with an attribute which represents its opinion, and the opinion of an agent changes because of interactions with other agents, without supervised coordination. Each interaction involves only two agents, and it corresponds to an exchange of messages. The framework assumes that time is modeled as a sequence of discrete steps, which do not necessarily have the same duration, and that at each step two random agents interact. Various sociological phenomena can be incorporated in the proposed framework, and the framework allows studying macroscopic properties of a system starting from microscopic models of such phenomena, obtaining analytic results. In detail, the proposed framework is inspired by the kinetic theory of gas mixtures, which relies on the use of balance equations that can be properly adopted to study opinion dynamics in a multi-agent system. After a short introduction on the kinetic theory of gas mixtures, this paper shows how the main ideas behind it can be generalized to study the dynamics of the opinion in multi-agent systems starting from a detailed description of microscopic interactions among agents.

1 Introduction

In this paper, we outline the presentation of an analytical framework to study collective behaviors in multi-agent systems without supervised coordination. According to the proposed framework, observable macroscopic properties of a system can be analytically derived, under proper assumptions, from the description of the effects of microscopic interactions among agents. The term *interaction* is frequently used throughout the paper to denote a message exchange among two agents, and each interaction corresponds to a single time step. Hence, time is modeled as a sequence of discrete steps, which may not have the same duration, and each step corresponds to a single interaction among a randomly chosen pair of agents. We assume that each agent is associated with a scalar attribute, and

* The work presented in this paper is partially supported by Gruppo Nazionale per il Calcolo Scientifico (GNCS).

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_37

since the scenarios considered in this paper refer to the study of opinion dynamics, we assume that such an attribute represents an agent's opinion. Most of the existing agent-based models used to study opinion dynamics are based on simulations and, hence, the validity of obtained results depends on the specific type of system that is simulated, and on the actual values of the parameters of simulations. At the opposite, the framework outlined in this paper adopts an analytic point of view, thus leading to results which are valid regardless of the details of simulations and of the number of agents in the studied multi-agent system, provided that the hypotheses used to derive the analytic results remain valid. Notably, the ideas behind the proposed framework are not limited to the study of opinion dynamics and they could be reworked to describe other attributes and other collective behaviours.

In detail, the analytic framework described in this paper to study opinion dynamics is inspired by the classic models that physics uses to study gases. This idea is not new, and other models used to study phenomena like opinion dynamics are derived from physics. For instance, it is worth mentioning models based on statistical mechanics [1] and on Brownian motion [2]. Besides these, other models of opinion dynamics are closely related to the proposed framework (see, e.g., [3]) because they are also inspired from the kinetic theory of gases, a branch of physics according to which macroscopic properties of gases can be explained starting from the details of microscopic interactions among molecules. The similarities between the kinetic theory of gases and the study of opinion dynamics are evident because the study of opinion dynamics typically starts from the description of microscopic interactions among agents and aims at deriving observable properties of the system concerning, e.g., the temporal evolution of the average opinion [4,5]. Due to such similarities, a simple parallelism between the molecules in gases and the agents in a multi-agent system can be drawn. This parallelism supports the derivation of kinetic-inspired analytical models to study opinion dynamics in multi-agent systems. Observe that classic kinetic theory of gases assumes that all molecules are equal. Hence, when generalizing the ideas of classic kinetic theory to study opinion dynamics, it is not possible to account for agents with different characteristics. In order to describe multi-agent systems in which agents may have different characteristics, the framework outlined in this paper extends those inspired by classic kinetic theory of gases. More precisely, instead of considering classic kinetic theory of gases, the discussed framework starts from kinetic theory of gas mixtures, which accounts for gases composed of different types of molecules. This allows modelling multi-agent systems composed of agents with different characteristics, thus adopting a parallelism between different types of molecules and different classes of agents.

Note that only some results of kinetic theory of gas mixtures are directly applicable to the study of opinion dynamics because the details of collisions among molecules intrinsically differ from those of interactions among agents. This is evident when comparing the interaction rules used to model collisions among molecules with the various interaction rules that can be considered in the study of opinion dynamics to account for various sociological phenomena

occurring in the interactions among agents. The following list enumerates, using the nomenclature proposed by sociologists, some of the most common sociological phenomena that can be accommodated in the analytic framework outlined in this paper by means of specific interaction rules [6–8]:

- *Compromise:* the tendency of agents to move their opinions towards those of agents they interact with, trying to reach consensus [9];
- *Diffusion:* the phenomenon according to which the opinion of each agent can be influenced by the social context [10];
- *Homophily:* the process according to which agents interact only with those with similar opinions [11];
- *Negative Influence:* the idea according to which agents evaluate their peers, and they only interact with those with positive scores [12];
- *Opinion Noise:* the process according to which a random additive variable may lead to arbitrary opinion changes with small probability [13]; and
- *Striving for Uniqueness:* the process based on the idea that agents want to distinguish from others and, hence, they decide to change their opinions if too many agents share the same opinion [14].

This paper is organized as follows. Section 2 outlines the basic nomenclature of the kinetic theory of gas mixtures. Section 3 presents the proposed framework using the nomenclature of kinetic theory of gas mixtures, and briefly discusses analytic results obtained considering only compromise. Finally, Section 4 concludes the paper and outlines ongoing and future work.

2 Overview of Kinetic Theory of Gas Mixtures

This section outlines the main ideas of the kinetic theory of gas mixtures, a branch of physics which was introduced to to allow modelling gases composed of different kinds of molecules, typically called *species*. Let $M \geq 1$ be the number of species in the considered gas. Then, M distribution functions $\{f_s(\underline{x}, \underline{v}, t)\}_{s=1}^{M}$, representing the density of molecules of species s at position $\underline{x} \in \mathbb{R}^3$ with velocity $\underline{v} \in \mathbb{R}^3$ at time $t \geq 0$, need to be considered, and a proper balance equation for each density function must be defined. Typically, the balance equation for distribution functions $\{f_s(\underline{x}, \underline{v}, t)\}_{s=1}^{M}$ is the Boltzmann equation and it is normally expressed as

$$\frac{\partial f_s}{\partial t} = \sum_{r=1}^{M} \mathcal{Q}_{sr}(f_s, f_r)(\underline{x}, \underline{v}, t) \qquad 1 \leq s \leq M. \tag{1}$$

The right-hand side of (1) represents the *collisional operator* relative to species s, which is meant to account for the details of interactions among agents. According to (1), such a collisional operator is written as the sum of the collisional operators $\mathcal{Q}_{sr}(f_s, f_r)$ relative to the collisions among the molecules of species s with the molecules of any species $r \in \{1, \ldots, M\}$. Observe that the collisional operator relative to species s does not only depend on the distribution function f_s, but it also depends on the distribution functions related to other species $\{f_r\}_{r=1, r\neq s}^{M}$.

Using the outlined analytic framework, proper macroscopic properties of the considered gas mixture can be studied, as follows. First, observe that the number of molecules of species s per unit volume at time t with position $\underline{x}$ is obtained by integrating the distribution function relative to species s with respect to $\underline{v}$

$$n_s(\underline{x}, t) = \int_{\mathbb{R}^3} f_s(\underline{x}, \underline{v}, t) \, \mathrm{d}_3\underline{v} \qquad 1 \leq s \leq M. \tag{2}$$

Then, according to the considered probabilistic approach, the average velocity of molecules of species s can be computed as

$$\underline{u}_s(\underline{x}, t) = \frac{1}{n_s(\underline{x}, t)} \int_{\mathbb{R}^3} \underline{v} \, f_s(\underline{x}, \underline{v}, t) \, \mathrm{d}_3\underline{v} \qquad 1 \leq s \leq M. \tag{3}$$

Finally, the temperature $T_s(\underline{x}, t)$ of molecules of species $s \in \{1, \ldots, M\}$ is computed according to the following formula

$$\frac{3}{2} n_s(\underline{x}, t) k T_s(\underline{x}, t) = m_s \int_{\mathbb{R}^3} |(\underline{v} - \underline{u}_s(\underline{x}, t))|^2 f_s(\underline{v}, \underline{x}, t) \, \mathrm{d}_3\underline{v} \qquad 1 \leq s \leq M \tag{4}$$

where parameters $\{m_s\}_{s=1}^M$ represent the mass of molecules of species s.

3 A Kinetic Framework for Opinion Dynamics

As discussed in the Introduction, it is possible to generalize the kinetic approach described in Section 2 to investigate opinion dynamics in multi-agent systems. According to described assumptions, we aim at studying interesting macroscopic properties of a system made of interacting agents, starting from the analysis of the effects of single interactions. While in kinetic theory the molecules of gases are associated with physical parameters, e.g., their positions and velocities, here we assume that each agent is associated with a single scalar parameter v which represents its opinion, and we assume that v varies continuously in a proper closed interval, denoted as $I \subseteq \mathbb{R}$. It is worth mentioning that, in the literature on opinion dynamics, opinion has been modeled either as a discrete [15] or as a continuous [16] variable. Discrete models are typically used to address situations where a finite number of possible options are available, e.g., in political elections. Continuous models, instead, are typically used to study opinions related to a single topic, varying from strongly disagree to completely agree.

In order to account for the peculiarities of the proposed framework, we call the groups of agents sharing the same characteristics *classes*, and not species, and we assume that the agents of the considered multi-agent systems are divided into $M \geq 1$ classes. It is possible to define M distribution functions $\{f_s(v, t)\}_{s=1}^M$, each of which represents the number of agents of a single class s with opinion in $(v, v + \mathrm{d}v)$ at time t. In agreement with the related equation of kinetic theory of gas mixtures, the number of agents of class s at time t can be written as

$$n_s(t) = \int_I f_s(v, t) \mathrm{d}v \qquad 1 \leq s \leq M. \tag{5}$$

The total number of agents is obtained by adding the number of agents of each class, as follows

$$n(t) = \sum_{s=1}^{M} n_s(t). \tag{6}$$

Moreover, following the same approach used in kinetic theory of gas mixtures, we can compute the average opinion of agents of class s as

$$u_s(t) = \frac{1}{n_s} \int_I f_s(v,t) v \mathrm{d}v \qquad 1 \leq s \leq M. \tag{7}$$

Observe that (7) is the analogous of (3), which rules the average velocity of each species in a gas mixture. The definition of the average opinions of all classes allows computing $u(t)$, i.e., the global average opinion of the system at time t

$$u(t) = \frac{1}{n(t)} \sum_{s=1}^{M} n_s(t) u_s(t). \tag{8}$$

Observe that the average opinion of the system is obtained by a properly weighed sum of the average opinions of all the classes. Finally, the standard deviation of the opinion relative to agents of class s can be computed as

$$\sigma_s^2(t) = \frac{1}{n(t)} \int_I (v - u_s)^2 f_s(v,t) \mathrm{d}v \qquad 1 \leq s \leq M. \tag{9}$$

Observe that (9) is related to (4), which is used to compute the temperature of each species in a gas mixture.

In the proposed framework, we assume that each distribution function $f_s(v,t)$ evolves according to a balance equation, which is the analogous of the Boltzmann equation (1)

$$\frac{\partial f_s}{\partial t}(v,t) = \sum_{r=1}^{M} \mathcal{Q}_{sr}(f_s, f_r) \qquad 1 \leq s \leq M. \tag{10}$$

From (10) it is evident that the collisional operator relative to class s corresponds to the sum of the contributions $\mathcal{Q}_{sr}(f_s, f_r)$ of the collisions between an agent of class s with an agent of class $r \in \{1, \ldots, M\}$.

As usual in the studies of the kinetic theory of gas mixtures, the Boltzmann equation is often treated in its *weak form* to try to obtain interesting analytic results. Actually, the study of the weak form of the Boltzmann equation is useful to derive proper differential equations involving the temporal evolution of interesting macroscopic parameters of the system, namely the average opinion of each class and the variance of the opinion of each class. The weak form of (10) is obtained by multiplying it by a so called *test function* $\phi(v)$ and by integrating with respect to v. Hence, the weak form of the Boltzmann equation for a given test function can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_I f_s(v,t) \phi(v) \mathrm{d}v = \sum_{r=1}^{M} \int_I \mathcal{Q}_{sr}(f_s, f_r) \phi(v) \mathrm{d}v \qquad 1 \leq s \leq M. \tag{11}$$

Proper choices of the test function allow obtaining analytic results regarding:

1. The number of agents of each class, using test function $\phi(v) = 1$;
2. The average opinion of each class, using test function $\phi(v) = v$; and
3. The variance of the opinion of each class, using test function $\phi(v) = (v - u_s)^2$.

In order to exemplify the analytic results that can be obtained using the outlined framework (see, e.g., previous works [6–8,17–20]), we briefly summarize major results obtained in the study of a very simple model of opinion dynamics which, among the variety of sociological phenomena cited in the introduction, considers only compromise. Let us remark that compromise is the key ingredient of many models of opinion dynamics and it is based on the idea that the opinions of two agents get closer after an interactions. Mathematically, denoting as v and w the pre-interaction opinions of two agents, and as v' and w' their post-interaction opinions, compromise corresponds to the following inequality

$$|v' - w'| < |v - w|. \tag{12}$$

In detail, compromise is modelled using the following interaction rules. Let us denote as s and r the generic classes of two interacting agents whose pre-interaction opinions are v and w, respectively. The post-interaction opinions v' and w' of the two agents are computed as

$$\begin{cases} v' = v - \gamma_{sr}(v - w) \\ w' = w - \gamma_{rs}(w - v) \end{cases} \tag{13}$$

where $\{\gamma_{sr}\}_{s,r=1}^{M}$ are the characteristic parameters of the model. The value of a single γ_{sr} quantifies the propensity of an agent of class s to change its opinion in favor of the opinion of an agent of class r. We remark that in order to reproduce compromise, namely to satisfy (12), the values of γ_{sr} must be chosen in the interval $\mathcal{I}_\gamma = (0, 1)$. As a matter of fact, using (13) it is possible to observe that (12) is equivalent to

$$|1 - (\gamma_{sr} + \gamma_{rs})| < 1 \tag{14}$$

which is always satisfied if all the parameters $\{\gamma_{sr}\}_{s,r=1}^{M}$ are defined in $\mathcal{I}_\gamma$. Using interaction rules (13), the weak form of the Boltzmann equation for each class $s \in \{1, \ldots, M\}$ can be explicitly written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_I f_s(v,t)\phi(v)\mathrm{d}v = \beta\sum_{r=1}^{M}\int_{I^2} f_s(v,t)f_r(w,t)(\phi(v^*) - \phi(v))\mathrm{d}v\mathrm{d}w \tag{15}$$

where on the left-hand side we used the fact that for every test function

$$\int_I \frac{\mathrm{d}}{\mathrm{d}t} f_s(v,t)\phi(v)\mathrm{d}v = \frac{\mathrm{d}}{\mathrm{d}t}\int_I f_s(v,t)\phi(v)\mathrm{d}v \qquad 1 \leq s \leq M. \tag{16}$$

Under these assumptions, considering the system of differential equations (11) with $\phi(v) = 1$, it can be proved that the number of agents of each class is conserved and, according to (6), the total number of agents is also conserved [7]. Moreover, from the analysis of the weak form of the Boltzmann equation (11) with $\phi(v) = v$, it is possible to show that the asymptotic average opinions of all classes are equal [21].

4 Conclusions and Future Work

This paper briefly outlines an analytic framework to study the dynamics of the opinion in multi-agent systems. In the studied scenarios, each agent is associated with a scalar parameter which represents its opinion and which may vary continuously in a closed interval. The opinion of each agent is modified by interactions with other agents. Each interaction corresponds to a message exchange between two randomly chosen agents, and it is assumed that each agent can freely interact with any other agent, without supervised coordination. In detail, the analytic framework proposed in this paper is inspired by the kinetic theory of gas mixtures, and it allows deriving analytic results which only rely on the choice of the details used to describe the effects of interactions among agents. Notably, the outlined framework draws a strong parallelism between the kinetic theory of gas mixtures and the study of opinion dynamics in multi-agent systems, even though it is worth remarking that analytic results regarding the dynamics of the opinion are different from those of the kinetic theory of gas mixtures because they are obtained using different forms of interactions.

Various interaction rules can be accommodated in the proposed framework and they can be used to account for some of the most commonly studied sociological phenomena involved in opinion dynamics. For the sake of brevity, this paper only focused on one of such phenomena, namely compromise, which represents the tendency of individuals to move their opinions towards those of the peers they interact with. Further work on the kinetic study of opinion dynamics involves treating other sociological phenomena, such as diffusion and negative influence, which have not yet been studied using the framework of gas mixtures. Another interesting problem that we plan to study concerns the modelling of locality of interactions, namely the fact that only agents close to each other are allowed to interact in some interesting models of opinion dynamics.

References

1. D. Stauffer and T. J. P. Penna, "Crossover in the Cont-Bouchaud percolation model for market fluctuations," *Physica A: Statistical Mechanics and Its Applications*, vol. 256, pp. 284–290, 1998.
2. F. Schweitzer and J. Holyst, "Modelling collective opinion formation by means of active brownian particles," *European Physical Journal B*, pp. 723–732, 2000.
3. G. Toscani, "Kinetic models of opinion formation," *Communications in Mathematical Sciences*, vol. 4, pp. 481–496, 2006.
4. W. Weidlich, *Sociodynamics: A systematic approach to mathematical modelling in the social sciences.* Amsterdam: Harwood Academic Publisher, 2000.
5. L. Pareschi and G. Toscani, *Interacting Multiagent Systems: Kinetic Equations and Montecarlo Methods.* Oxford: Oxford University Press, 2013.
6. S. Monica and F. Bergenti, "A study of consensus formation using kinetic theory," in *Proceedings of the* 13^{th} *International Conference on Distributed Computing and Artificial Intelligence* (DCAI 2016), Sevilla, Spain, June 2016, pp. 213–221.

7. F. Bergenti and S. Monica, "Analytic study of opinion dynamics in multi-agent systems with two classes of agents," in *Proceedings of* 17^{th} *Workshop "Dagli Oggetti agli Agenti"* (WOA 2016), Catania, Italy, July 2016.
8. S. Monica and F. Bergenti, "Opinion dynamics in multi-agent systems: Selected analytic models and verifying simulations," *Computational and Mathematical Organization Theory*, accepted September 2016.
9. R. P. Abelson, *Contributions to Mathematical Psycology.* New York: Frederiksen N., Gulliksen H. eds., 1964.
10. E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems." *Proc. Natl. Acad. Sci*, pp. 7280–7287, 2002.
11. A. Nowak, J. Szamrej, and B. Latan, "From private attitude to public opinion: A dynamic theory of social impact," *Psycol. Rev.*, vol. 97, pp. 362–376, 1990.
12. M. Mäs and A. Flache, "Differentiation without distancing. Explaining bipolarization of opinions without negative influence," *PLOS One*, vol. 8, 2013.
13. S. Galam, Y. Gefen, and Y. Shapir, "Sociophysics: A new approach of sociological collective behavior," *Journal of Mathematical Sociology*, 2009.
14. M. Mäs, A. Flache, and D. Helbing, "Individualisazion as driving force of clustering phenomena in humans," *PLOS One*, vol. 6, no. 10, 2010.
15. E. Yildiz, A. Ozdaglar, D. Acemoglu, A. Saberi, and A. Scaglione, "Noisy continuous opinion dynamics," *Journal of Statistical Mechanics*, vol. 9, pp. 1–13, 1982.
16. R. Hegselmann and U. Krause, "Opinion dynamics and bounded confidence models, analysis, and simulation," *Journal of Artificial Societies and Social Simulations*, vol. 5, no. 3, 2002.
17. S. Monica and F. Bergenti, "Simulations of opinion formation in multi-agent systems using kinetic theory," in *Proceedings of* 16^{th} *Workshop "Dagli Oggetti agli Agenti"* (WOA 2015), Napoli, Italy, June 2015.
18. S. Monica and F. Bergenti, "A kinetic study of opinion dynamics in multi-agent systems," in *Atti del Convegno* AI*IA 2015, Ferrara, Italy, September 2015.
19. S. Monica and F. Bergenti, "Kinetic description of opinion evolution in multi-agent systems: Analytic model and simulations," in *Proceedings of the* 18^{th} *International Conference on Principles and Practice of Multi-Agent Systems* (PRIMA 2015), Bertinoro, Italy, October 2015, pp. 483–491.
20. S. Monica and F. Bergenti, "An analytic study of opinion dynamics in multi-agent systems with additive random noise," in *Atti del Convegno* AI*IA 2016, Genova, Italy, December 2016.
21. S. Monica and F. Bergenti, "An analytic study of opinion dynamics in multi-agent systems," *Computer and Mathematics with Applications*, accepted March 2017.

Automatic Scheduling of Dependency-Based Workflows

Tahir Majeed, Michael Handschuh, and René Meier

Lucerne University of Applied Sciences and Arts, Mobile and Smart Systems Group, Rotkreuz, Switzerland
{tahir.majeed,michael.handschuh,rene.meier}@hslu.ch

Abstract. The distribution of energy is central to the needs of our society. The complexity of the tasks associated with maintaining energy distribution networks has increased exponentially with the introduction of renewable energy sources in large numbers. This paper proposes a novel approach for automatic optimization and scheduling of the workflows for energy distribution. The generated workflows replace manual scheduling of maintenance activities and consist of a set of inter-dependent tasks that have to be carried out by teams of engineers at different sites distributed across cities and even regions. Our algorithm is able to satisfy the dependencies between tasks and schedules these dependency based workflows by combining and extending on Job-Shop Scheduling and Traveling Salesman approaches. It minimizes the makespan of the workflows as well as the time teams travel to the various locations. The proposed approach has been assessed on different problem sizes and configurations based on data derived from real maintenance scenarios. Results show that high quality optimal solutions can be achieved for up to 21 sites within an hour.

Keywords: discrete optimization, workflow scheduling, job-shop, traveling salesman, mixed integer programming, energy distribution

1 Introduction

Electrical power is supplied to households in a city through electrical networks which consists of cable lines and grid stations. There are multiple routes through which power can be supplied to a household. These multiple routes ensure that power can be supplied uninterrupted even if some section of the network breaks down. To keep the whole network stable some of the lines are open (*i.e.* electricity cannot flow through them) while others are closed (*i.e.* electricity can flow through them). The configurations of these networks need to be changed frequently to guarantee supply of energy to all consumers at all time. In order to change a network configuration, a series of electrical Switching Task (ST) needs to be carried out. These ST are carried out inside grid stations located at different locations in a city. These ST are either switch on (close line) or switch off (open line) a line. Even in today's inter-connected world, a vast majority of

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_38

these grid stations need to be operated manually. Hence, a team of engineers has to physically travel to a grid station to carry out the ST. ST need to be carried out in a specific predefined sequence to ensure that a network stays in stable configuration and can continuously provide energy. Adherence to these sequences is furthermore paramount to guaranteeing the safety of the engineers that are working in this environment.

A set of ST combined together composes a workflow. If a workflow contains many ST then several teams are sent out simultaneously and the teams have to obey the dependencies in the workflow. At present the scheduling of the workflow is done manually by an experienced engineer who has through knowledge and deep understanding of the electrical switching network. However, with the increase of the network size due to growing population and due to the introduction of distributed renewable energy sources, such as solar power and wind energy, the complexity of the workflows has grown beyond manual scheduling. Automated approaches are needed to schedule these tasks into optimal workflows while guaranteeing the safety of the engineers and the continuous energy supply to consumers.

This paper proposes a novel approach to supporting automatic scheduling of dependency-based workflows that minimizes the makespan of workflows. This can be achieved by accomplishing two sub-objectives namely carrying out all tasks as early as possible and minimizing commuting time between the locations of the different grid stations. Since carrying out the tasks as early as possible depends on commuting time, minimizing commuting time will directly affect ST which can thus be carried out earlier.

Our approach extends on two different problems in the discrete optimization domain. Scheduling the ST relates to the Job-Shop Scheduling Problem (JSP) [10,13] while that of minimizing the travel time between grid stations relates to the Traveling Salesman Problem (TSP) [11]. As there can be several teams, the problem actually refers to the multiple Traveling Salesman Problem (mTSP) [7] where grid stations are visited in some specific order. The mTSP formulation will help minimize traveling time between different sites and the JSP formulation will help in scheduling the different ST. Both formulations must satisfy the dependency constraints.

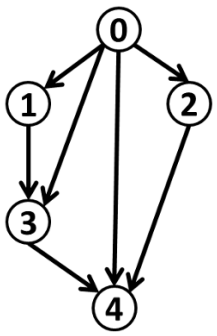
2 Related Work

The JSP is a classic problem that has been extensively studied during the last six decades. The JSP can be defined as assigning a finite number of resources called machines to a finite number of tasks called jobs over time with the aim of optimizing a given objective [12]. The objective can be achieved by either minimizing or maximizing a cost function which can be monetary or time-dependent. Typically, the jobs are independent of each other. JSP belongs to the class of NP-hard problems [16]. Many different approaches have been proposed over the years for solving JSP which includes meta-heuristic approaches *e.g.* evolutionary algorithms, tabu search, simulated annealing, ant colony optimization, hybrid

approaches and constraint directed techniques (see Özgüvena *et al.* [12], Šeda[16] and references therein). Others have used mathematical programming formulations to solve the JSP [10,14]. An elaborate review of the advancement in domain over the years is given by [4,8,9]. The problem addressed in this paper consists of jobs that are dependent on each other which has to be scheduled satisfying the dependencies and optimizing the makespan of the jobs.

The TSP has also been extensively studied during the last few decades and is defined as given a set of cities all of which have to be visited exactly once by a salesman. The salesman may visit cities in any order but should return to the starting city. The objective is to minimize the total distance traveled by the salesman [5,11]. A variant of TSP is the multiple Traveling Salesman Problem (mTSP) [7,15] where m-salesmen have to visit the cities and any of the m-salesmen can visit a city. Bektas [1] gives an overview of different integer programming formulations for mTSP.

The problem addressed in this paper is related to the one mentioned by Shapiro [14] but our proposed formulation is different from the one used by him. Our proposed approach fits naturally with the problem definition compared to the one proposed by Shapiro. He uses a 3-index formulation whereas our problem requires a 2-index formulation. He provides mathematical formulation for scheduling jobs on a single machine with changeover time which at an abstract level corresponds to a combination of TSP and JSP. Without providing a mathematical formulation, he mentions a possible solution for replacing a single machine with multiple machines and scheduling jobs on these machines. This paper provides complete mathematical formulation for a unified approach of mTSP and JSP.



ST	p_i
0	0
1	30
2	50
3	40
4	10

Fig. 1. Workflow

	0	1	2	3	4
0	0	0	0	0	0
1	1	0	0	0	0
2	1	0	0	0	0
3	1	1	0	0	0
4	1	0	1	1	0

Table 1. Dependencies

	0	1	2	3	4
0	0	100	144	85	76
1	100	0	60	140	153
2	144	60	0	152	175
3	85	140	152	0	32
4	76	153	175	32	0

Table 2. Distance matrix

3 Workflow Scheduling

As the computational power of computers increase over time along with the introduction of parallel processing on cluster machines and with the develop-

ment of more efficient Integer Programming libraries (*e.g.* Gurobi and CPLEX), increasing numbers of researchers are using mathematical programming-based scheduling [2]. A large class of optimization problem fits naturally into the mathematical programming paradigm where the problem constraints and the objective can be easily modeled and an optimizer can be used to provide an optimal or feasible solution.

The workflow scheduling of ST is casted as an optimization problem and mixed Binary Integer Programming (BIP) formulation has been chosen for optimization over heuristic based approaches. A BIP formulation can be easily adapted to changes in the problem objective and the constraints compared to heuristic based approaches. This scheduling problem is believed to be NP-hard, although a formal proof for the NP-hardness has yet to be computed. Experimental experience shows that it is much harder then the TSP even for small instances such as 22-sites and 2-teams.

The BIP formulation of the workflow scheduling is presented on a simple example shown in Fig. 1 which encapsulates the relevant aspects of larger and more complex workflows. The vertices in the directed graph represent a ST and the edges represent the dependencies between ST. ST are performed at a geographical location (henceforth referred to as sites), therefore, vertices also represent sites. The switching dependencies state that ST 4 (or ST at site 4) can not be performed unless and until ST 2 and 3 have been performed. These dependencies ensures that the problem is fundamentally different from mTSP where a team can visit any site. The dependencies are encoded by a dependency matrix as shown in Table 1. The table to the right in Fig. 1 shows the processing time required to complete the ST. In order to perform the ST at site 1, the team will first have to travel to site 1 and can carry out the associated ST only once they have reached the site. The matrix in Table 2 is a distance matrix and contains the times required to travel between the sites. Using the travel time instead of distances allows us to combine the travel time and the processing time in a single objective in the proposed mathematical formulation.

3.1 Mathematical Formulation

A workflow consists of a set of dependent ST $\mathbf{J} = \{j_1, j_2, \ldots, j_n\}$ with n tasks. Let $\mathbf{K}$ be the set of teams $\mathbf{K} = \{k_1, k_2, \ldots, k_m\}$ with m teams. Associated with each ST is a processing time $\mathbf{P} = \{p_i\}_{i=1}^{n}$ where p_i denotes the processing time required to complete ST j_i. Let $\mathbf{T} = \{t_1, t_2, \ldots, t_n\}$ where $t_i \in \mathbb{R}$ is the starting time of ST $i : i \in \mathbf{J}$. Let $\mathbf{D}$ be a binary 2D matrix of ST which represents the dependencies between the ST. If job j_i depends on job j_j then row i and column j of the matrix $\mathbf{D}$ is 1 and if job j_j is not dependent on job j_i then row j and column i of the matrix $\mathbf{D}$ is 0. The underlying dependency graph is a directed acyclic graph which ensures that there are no circular dependencies. Let $\mathbf{TD}$ be a set of pairs of ST that are either directly or transitively dependent. Let $\mathbf{ND}$ be a set of pairs of task that are not dependent on each other, *i.e.* there is no dependency constraint between these jobs and they can be performed before or after each other by the same team or in parallel to each other by different teams.

Let d_{ij} be the set of binary variables $d_{ij} = \{0, 1\}\ \forall i, j = 0, \ldots, n-1$ that is 1 if a team takes the route from site i to site j and 0 if a teams does not takes this route. Let a 2D matrix e encode the travel time required to travel between different sites Table 2. e_{ij} donates the time required to travel from site i to site j. The distance matrix is assumed to be symmetric $e_{ij} = e_{ji}$.

$$\text{Objective: min } \mathbf{Obj} = \sum_{i=1}^{n-1} t_i + \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} d_{ij} * e_{ij}. \tag{1}$$

$$\sum_{j=1}^{n-1} d_{0j} = m \tag{2}$$

$$\sum_{i=1}^{n-1} d_{i0} = m \tag{3}$$

$$\sum_{i=0}^{n-1} d_{ii} = 0 \tag{4}$$

$$\sum_{i=0, i\neq j}^{n-1} d_{ij} = 1 \qquad \forall j = 1 \ldots n-1 \tag{5}$$

$$\sum_{j=0, j\neq i}^{n-1} d_{ij} = 1 \qquad \forall i = 1 \ldots n-1 \tag{6}$$

$$d_{ij} + d_{ji} \leq 1 \qquad \forall 1 \leq i < j \leq n-1 \tag{7}$$

The following assumptions are made: the processing times are known and fixed; all the independent jobs are available at time zero. To make the integer programming formulation easier, a dummy ST is added at the Head Quarter (HQ) form where all the teams start with a processing time 0 (see Fig. 1). All the other ST are dependent on the dummy ST at the HQ. HQ is designated as site number 0. If a route d_{ij} is written as d_{0j} then this denotes a route from HQ to any site j and a route d_{i0} denotes a route form any site i to HQ. A team can perform only one ST at a time and each ST has to be performed only once. All teams are equally capable of performing the ST in the same time. The objective function **Obj** given by Eq. 1 can be defined for minimizing the starting time t_i of all the ST and the travel time $d_{ij} * e_{ij}$ which in turn is equivalent to minimizing the makespan of the workflow.

The constraints given by Eq. 2 through 6 are mTSP constraints (see Bektas [1]). Equations 2 and 3 ensure that m teams leave and return to HQ respectively. Equation 4 ensures that the route back to the same site is blocked. Equations 5 and 6 ensure that exactly one team enters and leaves a site respectively. Equation 7 is a novel constraint and ensures that teams do not return to the same site from where they left. This constraint ensures that a team either takes route d_{ij} or route d_{ji} or no route at all. It is important to mention that the sub-tour elimination constraints that are an integral part of mTSP formulation have been omitted in this formulation as they are not required. The sub-tour elimination constraints are indirectly enforced by the dependency constraint which are presented next.

$$t_i + p_i + (d_{ij} * e_{ij}) \leq t_j\ \ \forall \{i, j\} \in \mathbf{TD} \tag{8}$$

Equation 8 ensures that the tasks are performed in an order that satisfies the dependencies. The constraint ensures that if ST j is dependent on ST i then t_j is

greater than or equal to the sum of starting time of ST t_i, its processing time p_i and optionally if the same team that has performed ST i is going to perform ST j (*i.e.* $d_{ij} = 1$) then the travel time between the sites e_{ij}. This constraint is called precedence constraint and is widely used. It has been modified by incorporating the $(d_{ij} * e_{ij})$ term so that the mTSP and JSP formulations can be merged together. If a different team that has performed ST i is going to perform ST j then $d_{ij} = 0$ and t_j is then the sum of t_i and p_i.

$$t_i + p_i + e_{ij} + M * d_{ij} \leq M + t_j \ \ \forall\{i, j\} \in \mathbf{ND} \tag{9}$$

Equation 9 provides the constraint needed to serialize the tasks that are independent of each other and are called non-interference constraint. The non-interference constraint ensures that independent tasks are not performed simultaneously by the same team. The tasks however, can be performed simultaneously by different teams as there is no dependency between them. This constraint is inspired from the work of Kara and Derya [5] and has been modified to suit our problem. The constraint Eq. 9 ensures that if $d_{ij} = 1$ (*i.e.* same team is performing ST i and j) then t_j is equal to or greater than the sum of t_i, p_i and e_{ij}. If $d_{ij} = 0$ (*i.e.* different teams are performing ST i and j) then the constraint switches off and the ST i and j can be performed independently. The constraints given by Eq. 8 and 9 combine the mTSP and JSP problems together. The M (Big-M) used in the equation is a large enough number and is generally used to switch off the constraint. Equation 9 gives an indication of the lower bound on Big-M which is that it should be greater than $t_i + p_i + e_{ij} - t_j \leq M$. Assuming if $t_j = 0$ then $t_i + p_i + e_{ij} \leq M$. The largest value of t_i is for the task that is performed last. As we are minimizing t_i, naively computing the maximum tour through all the sites will provide an upper bound on the value of t_i. It is computed by visiting the site that is farthest from the current site and adding it to the route. The time it takes to reach the last site added to the processing time of all the tasks provides a lower bound on the value of Big-M.

To decrease the search space and find good solutions as early as possible some heuristics were employed. The classical TSP assumes that the salesman can visit any site in any order. However, for the given problem teams may not visit the sites in arbitrary order. Making use of this dependency structure, if a ST j is dependent on ST i then a team is not allowed to travel from site j to site i to perform ST i after ST j. If such routes were allowed then it would violate the dependencies. The same is true for all the direct and transitive dependencies. Therefore, to reduce the number of variables and to disallow such routes, considering all the dependency pairs in the set **TD**, the corresponding d_{ji} variables were set to 0 where task j is dependent on task i.

4 Results and Discussion

The model presented above was coded in C++ and Gurobi 7.0 [3] was used as the optimizer. The model was tested on an Intel(R) Xeon(R) CPU E5540 with 2-Core 2.53GHz and 32GB RAM running Windows 8.1 Enterprise 64-bit.

The model was tested on different datasets derived from real world switching scenarios for two to five teams and the results are presented in Table 3. Table 3 shows the number of sites (Sites), number of teams (Teams), whether the solution was optimal or feasible (Status), objective value (Obj), percentage of travel time (TT) and percentage of task start time (PT) in the objective value, in percentage the gap between the upper and lower bound (Gap), and the run time (Time) of the solver until it found the optimal or feasible solution. The computational complexity of the decision variables is $\mathcal{O}(n^2)$. The Gurobi solver was able to find optimal solution for up to 21 sites within the stipulated time which was set to 10 hours. The solver was unable to prove the optimality of the found solution within the stipulated time frame for higher number of sites (see Table 3 22-sites and 2-teams), however, it was able to find high quality solution even in this case. Through experimental experience we believe that this is also the optimal or close to optimal solution but the solver requires more time to prove the optimality. Figures 2 and 3 shows the color coded optimized route and schedule for 12 sites and 3 teams respectively. Our experiments demonstrate that for a higher number of teams the solution is found earlier as compared to smaller number of teams as can be seen in Table 3. We believe this is due to a quicker decrease in search space with more teams compared to less number of teams. However, we plan to investigate this further in future work.

Table 3. Result with different number of sites and teams.

Sites	Teams	Status	Obj	TT(%)	PT(%)	Gap(%)	Time(sec)
12	2	Optimal	3262	24.6	75.4	0.0	3.5
12	3	Optimal	2494	34.2	65.8	0.0	0.9
12	4	Optimal	2455	41.8	58.2	0.0	1.0
12	5	Optimal	2463	44.1	55.9	0.0	0.5
19	2	Optimal	20434	13.3	86.7	0.0	499.7
19	3	Optimal	15064	18.2	81.8	0.0	75.3
19	4	Optimal	12490	21.1	78.9	0.0	6.4
19	5	Optimal	11466	24.8	75.2	0.0	3.2
20	2	Optimal	18888	13.5	86.5	0.0	702.3
20	3	Optimal	13796	18.7	81.3	0.0	105.7
20	4	Optimal	11452	23.4	76.6	0.0	17.2
20	5	Optimal	10465	24.6	75.4	0.0	5.9
21	2	Optimal	19993	12.6	87.4	0.0	3169.6
21	3	Optimal	15264	18.1	81.9	0.0	846.9
21	4	Optimal	12750	21.5	78.5	0.0	77.4
21	5	Optimal	11642	24.7	75.3	0.0	35.2
22	2	Feasible	21937	12.3	87.7	>10.0	36000.0
22	3	Optimal	15324	17.8	82.2	0.0	6945.3
22	4	Optimal	12068	22.0	78.0	0.0	70.8
22	5	Optimal	11146	26.5	73.5	0.0	31.0

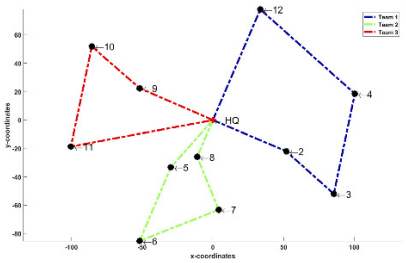


Fig. 2. TSP solution for 12-sites and 3-teams.

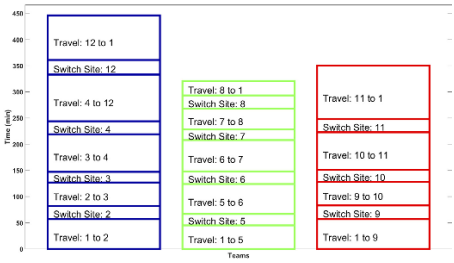


Fig. 3. JSP solution for 12-sites and 3-teams.

5 Conclusion and Future Work

A novel approach has been proposed for automatic optimization and scheduling of workflows. It builds upon JSP with inter-dependent tasks together with mTSP and has been designed for directed acyclic graphs. The proposed approach has been applied in the the domain of energy distribution. However, it is a general approach and can be applied whenever dependent tasks have to be scheduled which also involves traveling. Our results show that optimal solutions can be achieved within an hour even for larger networks. The proposed approach can be extended by incorporating additional features which are left for future work. For instance, the task processing time is assumed to be constant irrespective of the teams. In reality teams may be experienced or inexperienced and their processing times may differ, *e.g.* an experienced team will complete a task sooner than an inexperienced team. Based on our experience it should be feasible to reformulate the problem to account for team dependent processing time. The workflow may also be optimized for the optimal number of teams where each team incurs a certain cost. Kek *et al.* [6] provides a formulation for a vehicle routing problem that also optimizes the number of vehicles used.

Acknowledgments

This work was funded in part by the Swiss Commission for Technology and Innovation (CTI). The authors would also like to thank IDS Switzerland AG for their support and for providing the data that made this work possible.

References

1. Bektas, T.: The Multiple Traveling Salesman Problem: An Overview of Formulations and Solution Procedures. Omega 34(3), 209–219 (2006)
2. Chen, J.S., Yan, J.S.: Model Formulations for the Machine Scheduling Problem with Limited Waiting Time Constraints. Journal of Information & Optimization Sciences 27(1), 225–240 (2006)

3. Gurobi Optimization, I.: Gurobi Optimizer Reference Manual (2017), http://www.gurobi.com
4. Jain, A.S., Meeran, S.: Deterministic Job-Shop Scheduling: Past, Present and Future. Operational Research 113(2), 390–434 (1998)
5. Kara, I., Derya, T.: Formulations for Minimizing Tour Duration of the Traveling Salesman Problem with Time Windows. In: Procedia Economics and Finance. vol. 26, pp. 1026–1034 (2015)
6. Kek, A.G., Cheu, R.L., Meng, Q.: Distance-Constrained Capacitated Vehicle Routing Problems with Flexible Assignment of Start and End Depots. Mathematical and Computer Modelling 47(1), 140–152 (2008)
7. Laporte, G., Nobert, Y.: A Cutting Planes Algorithm for the m-Salesmen Problem. Journal of the Operational Research Society 31(11), 1017–1023 (1980)
8. Lawler, E.L., Lenstra, J.K., Kan, A.H.R., Shmoys, D.B.: Sequencing and Scheduling: Algorithms and Complexity. Handbooks in Operations Research and Management Science 4, 445–522 (1993)
9. Lee, C.Y., Lei, L., Pinedo, M.: Current Trends in Deterministic Scheduling. Annals of Operations Research 70(0), 1–41 (1997)
10. Manne, A.S.: On the Job-Shop Scheduling Problem. Operations Research 8(2), 219–223 (1960)
11. Miller, C.E., Tucker, A.W., Zemlin, R.A.: Integer Programming Formulation of Traveling Salesman Problems. Journal of the ACM 7(4), 326–329 (1960)
12. Özgüven, C., Özbakr, L., Yavuz, Y.: Mathematical Models for Job-Shop Scheduling Problems with Routing and Process Plan Flexibility. Applied Mathematical Modelling 34(6), 1539–1548 (2010)
13. Ronconi, D.P., Birgin, E.G.: Mixed-Integer Programming Models for Flowshop Scheduling Problems Minimizing the Total Earliness and Tardiness, vol. 60, pp. 91–105. Springer (2011)
14. Shapiro, J.F.: Handbooks in Operations Research and Management Science, vol. 4, chap. Mathematical Programming Models and Methods for Production Planning and Scheduling, pp. 371–443. Elsevier (1993)
15. Taş, D., Gendreaub, M., Jabalic, O., Laporte, G.: The Traveling Salesman Problem with Time-Dependent Service Times. European Journal of Operational Research 248(2), 372–383 (2016)
16. Šeda, M.: Mathematical Models of Flow Shop and Job Shop Scheduling Problems. International Journal of Applied Mathematics and Computer Sciences 4(4), 241–246 (2007)

Preliminary study for improving accuracy on Indoor positioning method using compass and walking detect

Takayasu Kawai[1], Kenji Matsui[1], Yukio Honda[1],
Gabriel Villarubia [2], Juan Manuel Corchado Rodriguez[2]

[1]Department of Engineering, Osaka Institute of Technology
1-45, Chayamachi, Kita-ku Osaka-shi, Osaka, 530-0013, Japan
m1m16h05@st.oit.ac.jp, kenji.matsui@oit.ac.jp
[2]BISITE Research Group, University of Salamanca,
Edificio I+D+I, 37008 Salamanca, Spain
{gvg, corchado}@usal.es

Abstract. Indoor positioning technology is commercially available now, however, the positioning accuracy is not sufficient in the current technologies. Currently available indoor positioning technologies differ in terms of accuracy, costs and effort, but have improved quickly in the last couple of years. It has been actively conducted research for estimating indoor location using RSSI (Received Signal Strength Indicator) level of Wi-Fi access points or BLE (Bluetooth Low Energy) tags. WiFi signal is commonly used for the indoor positioning technology. However, It requires an external power source, more setup costs and expensive. BLE is inexpensive, small, have a long battery life and do not require an external energy source. Therefore, using BLE tags we might be able to make the positioning system practical and inexpensive way. In this paper, we investigate such practical type of indoor positioning method based on BLE. BLE RSSI are processed by Multilayer Perceptron(MLP). Also, compass data and walking speed estimation with an extended Kalman filter is used to improve the accuracy. Our preliminary experimental result shows 2.21m error in case of the MLP output. In preliminary experimental results, the proposed approach improved the accuracy of indoor positioning by 21.2%.

Keywords: indoor positioning, BLE, fingerprint, Extended Kalman filter

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_39

1 Introduction

In case of outdoor positioning, satellite-based GPS positioning works very well. However, indoor positioning is not so straight forward, and that area is glowing research fields in mobile computing because of the popularization of mobile devices, like smartphones, tablets. The technologies currently used most in the development of Real-Time Location System are, RFID (Radio Frequency IDentification), Wi-Fi or BLE (Bluetooth Low Energy)[1][2][3][4]. Indoor positioning technology is commercially available now, however, the positioning accuracy is not sufficient in current technologies. Currently available indoor positioning technologies differ in terms of accuracy, costs and effort, but have improved quickly in the last couple of years. WiFi signal is commonly used for the indoor positioning technology. However, WiFi equipment requires an external power source, more setup costs and expensive. Bluetooth Low Energy (BLE) is one of the latest technologies. It is called BLE beacons (or iBeacons) that are inexpensive, small, have a long battery life and do not require an external energy source. BLE technologies bring a couple of advantages: easy deployment and are integrated in most of the current electronic devices. Thus, it has been actively conducted research for estimating indoor location using RSSI (Received Signal Strength Indicator) level of BLE tags. There are also some indoor localization studies using accelerometer and geomagnetic sensor. Most modern smartphones are equipped with those sensors. Those studies estimated both position and direction of the target object and also detected walking, but their positioning accuracy was insufficient. Kumar[5] reported localization using the RSSI values of the anchor node beacons and the multi-layer perceptron (MLP)-based classifier. Zhang[6] utilized smartphone sensors and Kalman filter for location estimation. Liu[7] also tried smartphone sensors and particle filters to improve the location estimation. In this paper, we report a preliminary study of position estimation using ibeacon RSSI and smartphone sensors (inertial and geomagnetic) combined with Extended Kalman Filter-based estimation. In terms of the calculation cost and extension to non-linear system, in this study, extended Kalman filter has been investigated for the estimation of the indoor location.

2 Proposed Method

Figure 1 shows the basic building block of the proposed positioning system. The inputs are the location information of BLE tags and RSSI. The MLP estimates the location where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the expected location coordinates. The advantages of using BLE tags are not only improving the cost performance, but we might be able to implement many applications associated with BLE tags. Using the smartphone, the acceleration information and the geomagnetic information are obtained and converted into the velocity and the rotational velocity values, respectively. Then the extended Kalman filter estimates the location (X, Y) using the inputs ($\boldsymbol{x}, \boldsymbol{y}$), velocity, and rotational velocity.

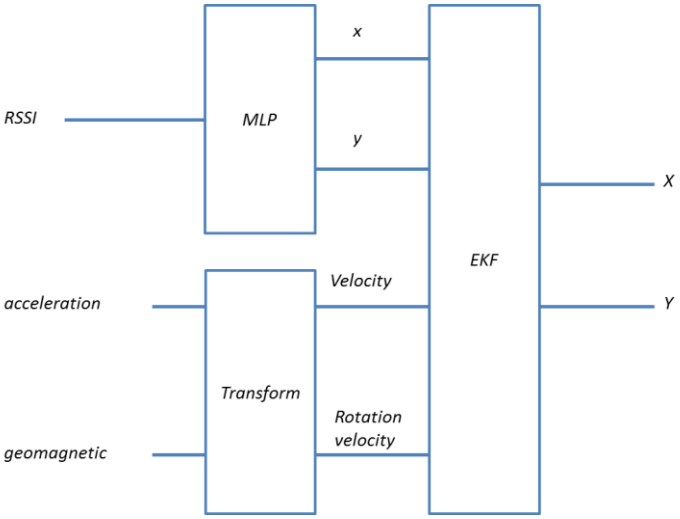


Figure 1. Block diagram of the proposed system

3 Location Estimation using MLP with Fingerprints

The estimation accuracy of the MLP output was evaluated. First, the training data of fingerprints were collected using the smartphone. The experimental environment is shown in Figure 2. Four BLE beacons were symmetrically arranged along the periphery of the (4 × 4)-m2 square zone. RSSI values were measured 20 times at each 16 blocks as shown in Figure 2. For the evaluation, we took the dotted route in Figure 2, and measured RSSI values from the four BLEs every 1sec. The location was then estimated by the MLP algorithm. The experimental result showed that the maximum error, the minimum error and the average error were 3.61m, 1.00m, 2.21m, respectively. Figure 3 shows the measured RSSI values.

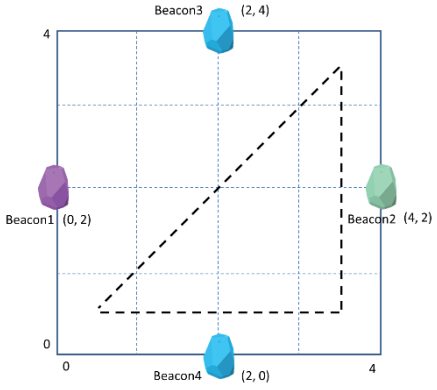


Figure 2. Experimental environment

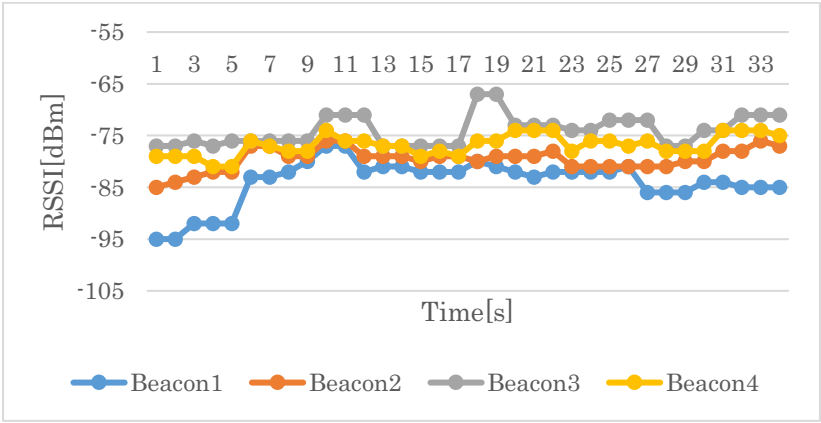


Figure 3. Observed RSSI values

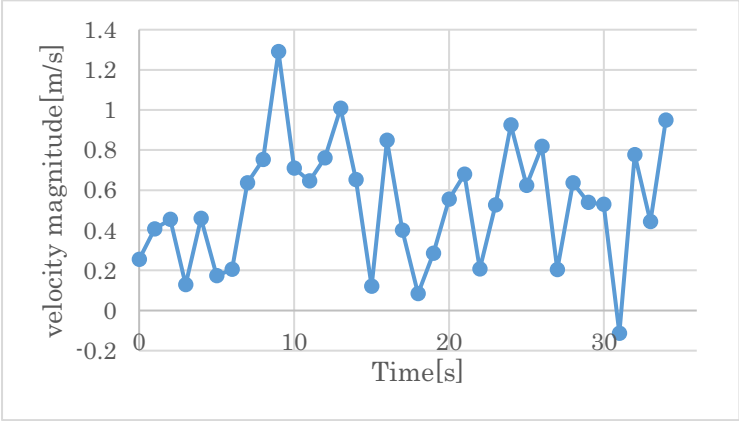


Figure 4. Observed velocity values

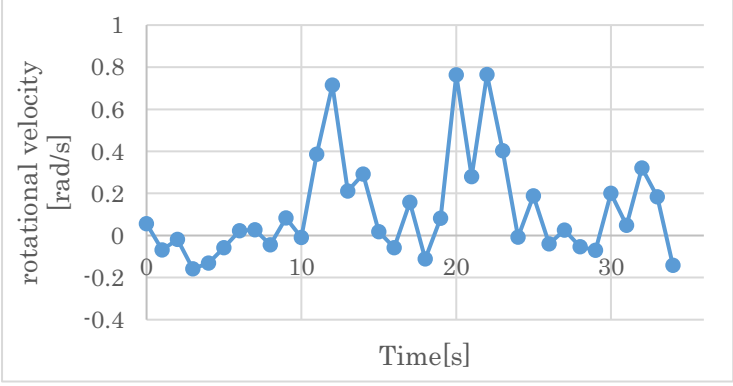


Figure 5. Observed rotation velocity values

Our simple and small area localization test using MLP and fingerprint aims to understand the basic performance using BLE tags. The result was not stable enough, however, we will use the estimation result as the reference to the proposed method.

4 Evaluation of the Proposed Method

Figure4 and Figure5 are the velocity and the rotation velocity respectively. Those values were measured using smartphone together with the RSSI values as shown in Figure3. In our proposed method, using extended Kalman filter, the discrete-time state transition model is as follows;

$$\boldsymbol{x}(k) = \boldsymbol{f}\big(\boldsymbol{x}(k-1), \theta(k-1), v(k-1)\big) + \boldsymbol{e_1}(k-1)$$

$$\boldsymbol{f}(\boldsymbol{x}\,(k-1), \theta(k-1), v(k-1)) = \begin{bmatrix} x(k-1) + Tv(k-1)\cos\theta(k-1) \\ y(k-1) + Tv(k-1)\sin\theta(k-1) \\ \theta(k-1) + T\omega(k-1) \\ v(k-1) \end{bmatrix}$$

where location coordinates (*x, y*), velocity v, attitude angle θ, rotational velocity ω, sampling period T were used. $\boldsymbol{e}_1(k)$ is the process noise which is assumed to be zero mean multivariate Gaussian noise with covariance Q. Also, the observation equation is as follows;

$$\boldsymbol{y}(k) = \begin{bmatrix} x_e(k) \\ y_e(k) \end{bmatrix} = \boldsymbol{h}(x(k)) + \boldsymbol{e_2}(k)$$

where the coordinates $(x_e(k), y_e(k))$ is obtained from the MLP outputs, $\boldsymbol{h}(\boldsymbol{x}(k))$ is a function representing the observation, $\boldsymbol{e}_2$ is the noise input to the observation and follows the covariance matrix R.

As shown below, Step1 and Step2 are the prediction and filtering steps of the extended Kalman filter, respectively. From the above state equation and observation equation, the extended Kalman filter is calculated by the following two-step recursive processing.

Step1

$$\hat{\boldsymbol{x}}^{-}(k) = \boldsymbol{f}(\hat{\boldsymbol{x}}(k-1), \theta(k-1), v(k-1))$$

$$\boldsymbol{P}^{-}(k) = \boldsymbol{F}(k)\boldsymbol{P}(k-1)\boldsymbol{F}^{\boldsymbol{T}}(k) + Q$$

Step2

$$\boldsymbol{g}(k) = \frac{\boldsymbol{P}^{-}(k)\boldsymbol{H}(k)}{\boldsymbol{H}^{\boldsymbol{T}}(k)\boldsymbol{P}^{-}(k)\boldsymbol{H}(k) + R}$$

$$\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}^{-}(k) + \boldsymbol{g}(k)\{\boldsymbol{y}(k) - \boldsymbol{h}(\hat{\boldsymbol{x}}^{-}(k))\}$$

$$\boldsymbol{P}(k) = \{\boldsymbol{I} - \boldsymbol{g}(k)\boldsymbol{H}^{T}(k)\}\boldsymbol{P}^{-}(k)$$

$$\boldsymbol{F}(k) = \frac{\partial \boldsymbol{f}(\boldsymbol{x}(k), \theta(k), v(k))}{\partial \boldsymbol{x}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -Tv(\mathrm{k})\sin\theta(k) & Tv(\mathrm{k})\cos\theta(k) & 1 & 0 \\ T\cos\theta(k) & T\sin\theta(k) & 0 & 1 \end{bmatrix}$$

where the ***H(k)*** is a 4 × 4 identity matrix, $\hat{\boldsymbol{x}}^{-}$ is the prestate estimate, $\hat{\boldsymbol{x}}$ is the state estimate, $\boldsymbol{P}^{-}$ is the previous error covariance matrix, $\boldsymbol{P}$ is the posteriori error covariance matrix, and $\boldsymbol{g}$ is the Kalman gain. By calculating these values, the estimated value can be obtained.

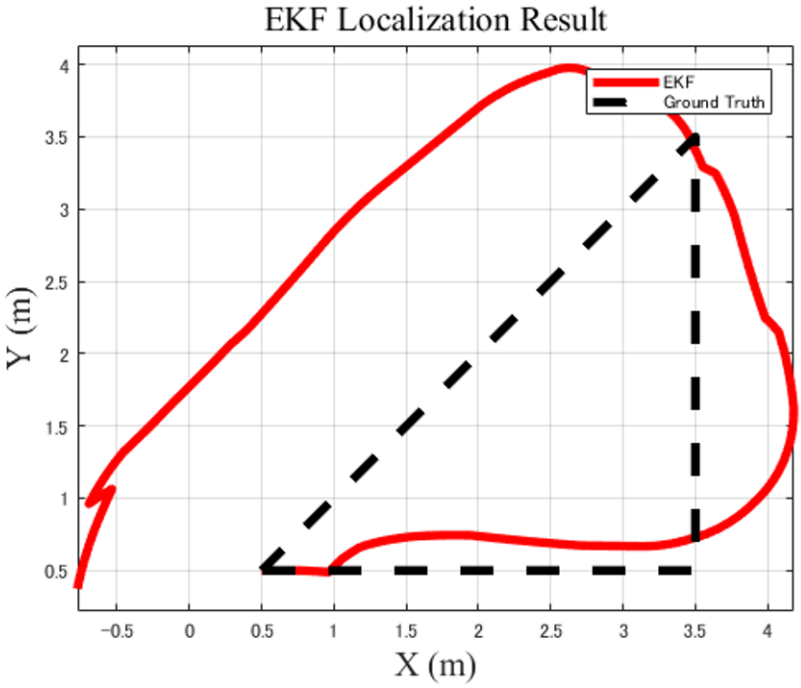


Figure 6. The actual walking path (dotted line) and the estimated result (red line)

5 Experimental Results

The experimental result showed that the maximum error, the minimum error, and the average error were 2.92m, 0.51m, and 1.75m. This experiment was measured thirty-four times in every one second. The average error, minimum error and maximum error were 21.2%, 49.0% and 19.1% lower compare with the result from the experiment using just MLP.

6 Discussion

The error is considerably smaller in case of the proposed method than the method using just MLP. However, the accuracy remains insufficient, and need to be carefully investigated. Also, the accuracy is depending on the experimental environment. Therefore, we need to consider the factors which make RSSI unstable. We will also investigate other estimation methods, which were not carefully considered in the present study.

7 Conclusion

The accuracy improvement of an indoor positioning method using a compass and walking detection was assessed in a preliminary study. MLP, linear velocity, and rotational velocity were combined and tested in an extended Kalman filter. We compared the overall fingerprint-matching performance between the proposed method and a conventional method. The accuracy was improved in the proposed method (by over 20%), but the improvement was insufficient.

8 Reference

1. Villarubia, G., Rubio, F., De Paz, J. F., Bajo, J., & Zato, C. (2013). Applying classifiers in indoor location system. In *Trends in Practical Applications of Agents and Multiagent Systems* (pp. 53-58). Springer International Publishing.
2. Kaemarungsi, K., & Krishnamurthy, P. (2004, March). Modeling of indoor positioning systems based on location fingerprinting. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies* (Vol. 2, pp. 1012-1022). IEEE.

3. Liu, H., Darabi, H., Banerjee, P., & Liu, J. (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *37*(6), 1067-1080.
4. Sala, A. S. M., Quiros, R. G., & Lopez, E. E. (2010, June). Using neural networks and Active RFID for indoor location services. In *Smart Objects: Systems, Technologies and Applications (RFID Sys Tech), 2010 European Workshop on* (pp. 1-9). VDE.
5. Kumar, S., & Lee, S. R. (2014, June). Localization with RSSI values for wireless sensor networks: An artificial neural network approach. In *International Electronic Conference on Sensors and Applications* (Vol. 1). Multidisciplinary Digital Publishing Institute.
6. Zhang, R., Bannoura, A., Höflinger, F., Reindl, L. M., & Schindelhauer, C. (2013, February). Indoor localization using a smart phone. In *Sensors Applications Symposium (SAS), 2013 IEEE* (pp. 38-42). IEEE.
7. Liu, Y., Dashti, M., Rahman, M. A. A., & Zhang, J. (2014, March). Indoor localization using smartphone inertial sensors. In *Positioning, Navigation and Communication (WPNC), 2014 11th Workshop on* (pp. 1-6). IEEE.

Recognition of Table Images Using K Nearest Neighbors and Convolutional Neural Networks

Ujjwal Puri[1] ujjwalpuri@gmail.com, Amogh Tewari[1] amogh_tewari@yahoo.co.in, Shradha Katyal[1] skatyal2204@gmail.com, and Dr.Bindu Garg[1] bindu.garg@bharatividyapeeth.edu

Bharati Vidyapeeth's College of Engineering, A 4, Rohtak Road, Paschim Vihar, Delhi, 110063 coedelhi@bharatividyapeeth.edu

Abstract. The objective of this research paper is to analyze images of tables and build a prediction system capable of recognizing the number of rows and columns of the table image with the help of Convolutional Neural Networks and K Nearest Neighbours. The data set used in the building of the models has been indigenously created and converted to gray-scale. The eventual objective and possible application of the paper is to assist the building of software capable of reading tables from non digital sources and creating digital copies of them.

1 Introducion

The conventional edge detection algorithms work majorly by detecting gradient shifts and other features of images, which can be a fairly intensive computational task for a hand-held device. Also, this computation needs to be redone for every image. Using a trained CNN is comparatively much less taxing on the processor while still achieving high accuracy.

There already exists handwritten text recognition tools with good accuracy on par with human recognition. This paper attempts to show that KNN and CNN can be used to detect the number of rows and columns of a table, which essentially means that these techniques can further be implemented to detect individual cells of a table. By incorporating the text recognition tools and our attempt of detecting rows and columns of a table, it becomes possible to read tables from photos.

2 Literature review

Pattern recognition in nature can be a daunting task, a tailor made algorithm for a particular set of pattern may completely break down for another, but a self learning solution that after a substantial amount of experience(data) can stand the test for multiple patterns[9]. Convolutional neural networks work on the above principle [8,7], they are very efficient at learning and identifying certain 2D variances in the provided data.
The structure of the Convolutional net effects the accuracy of the pattern recognition. For large scale image recognition the increase in the number of convolutional layers has

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_40

shown a significant increase in the accuracy achieved[12].
Nearest neighbour algorithms don't require pre-processing of the labels in most cases[2]. K nearest neighbours in general require large amounts of time to classify each test case due to the large amount of comparisons required when compared to other more sophisticated machine learning techniques[13]. Furthermore, different sorting and optimization algorithms can be used to speed up the classification process[5].
Algorithms based on Euclidian distances are prone to requiring exponential amounts of time for testing[11]. The nearest neighbour algorithms are most effective when the difference in distances between various labels is big enough[1]. However, if the problem is based on discrete values which is the basis of classification, the value of solving the nearest neighbour problem is retained. The difference between three rows and four columns, and vice versa might not be much for the nearest neighbour algorithm in case the lengths of the dividing lines are the same, but it is easy to recognise that the value of the difference between the two tables is more in the subjective sense of the identity of the tables being different.
Weighted nearest neighbour classification suggests that the weights to a nearest neighbour algorithm should be considered according to the distance between the sample and the known classified neighbour [4]. To avoid a non-bayes decision of a test sample remaining unclassified, it is advisable to use a k-nn instead of 1-nn[3,6]. Further work relating to using the k nearest neighbour algorithm in recognition of table images is possible using the KNNMP which is the k nearest neighbour moving point problem [14].

3 The Dataset

There were two datasets created one digitally and the other hand drawn. The digital dataset used to train the Convolutional Neural Network and the KNN consists of 1000 images of tables of varying sizes, with their corresponding number of rows and columns.
The dataset was created using a python script written by us. The script generated HTML tables of varying sizes ranging between 2×1 to 3×3 randomly. The cells of the tables were filled in with random characters ranging from ASCII code 32 to 125. The characters were not just restricted to the English alphabet but rather included most of the printable characters present on a US-English keyboard, this was done to increase the degree of randomness of the tables generated.

After the generation of tables, there were two options of capturing the images of the tables, either it could have been done by controlling a webcam to take images of the screen or by taking simple screenshots.
Using webcams posed many possible points of failures, the glare of the screen could have tainted the images, the placement of the webcam would have to be meticulously chosen, the image processing would also have been limited due to the already low capture quality, plus the process would be slower even if marginally than screenshots. The screenshot option was much cleaner, efficient and had more room for various image processing.
The screenshots were of the dimension 288×90 and were converted to grayscale as the

color component wasn't necessary for classifying the number of rows and columns in the tables. After down sampling the images to the dimension 96×30, an antialiasing filter was applied found in Python Image Library (PIL) and the grey intensities of the screenshots were normalized between 0 (white) and 1 (black).

aO%KvS%wq	wALtqCJL@P9c_
nIW0=4i&n	xKwp

Fig. 1. An example of the final image of the digitally produced table.

The hand drawn dataset contains a total of 180 tables, 30 for each dimension. The tables were drawn on paper and then scanned. These scanned tables were then put through the same image processing as the screenshot tables.

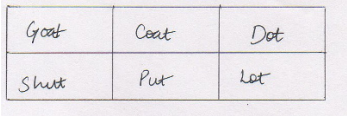


Fig. 2. An example of the final image of the hand drawn table.

4 The Convolutional Neural Networks Approach

The convolutional neural networks consist of three different main elements used to identify a test input: Local receptive field, Pooling Layer, Shared weights and biases. The overall basic structure can be seen in fig.3.

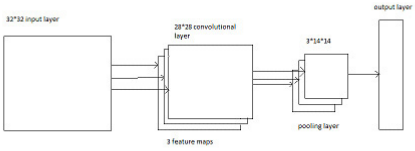


Fig. 3. The basic structure of the CNN used.

Our input layer was an n-d array of grey intensities of the images of the tables with resolution 96×30. Each neuron then in the hidden layer was connected to a region of

input neurons. This region known as the local receptive field can be varied to observe its effect on the final output. A feature map, maps the connection from the input layer to the hidden layer. The weights of this map are known as shared weights and the bias is called shared bias. The advantage of having shared weights and biases is that the number of parameters is reduced in comparison to a fully connected model. This results in convolutional networks being fast learners.

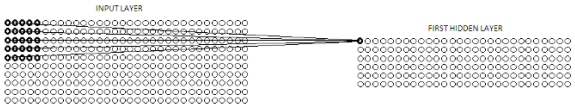


Fig. 4. Shows a local receptive field of 5×5 neurons in the input layer connected to the first hidden layer(Convolutional Layer).

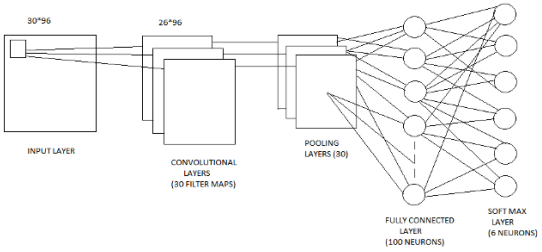


Fig. 5. The final CNN network used.

Pooling layers [10] are a condensed form of the convolutional layers. They take their input from the output of the convolutional layers. Each neuron in a pooling layer may be associated with a region of $m \times m$ neurons, where $m < n$. The regions do not overlap unlike local receptive fields. We've implemented max pooling.

4.1 Tackling over fitting

After the pooling layer there is a fully connected layer. In the fully connected layer we've applied the dropout technique [15]. The fully connected layer then is followed by a Soft-Max layer, in which instead of applying the same activation function as the previous layers, soft-max function is applied

$$a_j^L = \frac{e^{z_j^L}}{\sum_k e^{z_k^L}}$$

Where a is the activation value of the jth neuron in the lth layer with k neurons, and z is the weighted input($\sum_k w_{jk}^L a_k^{L-1} + b_j^L$).

The activation value of a neuron in the softmax layer is inversely proportional to the weighted input of all the neurons in the softmax layer.

4.2 Results for Convolutional Neural Net

The maximum accuracy achieved for digitally drawn tables was 99.79, which was achieved by all the three activation functions. Decreasing the batch size down to 20 caused a minor dip in the accuracy. Fig.6 shows that all the three nets of different activation functions reached the maximum accuracy fairly quickly, Sigmoid being the fastest and Tanh the slowest.
For the hand drawn dataset the maximum accuracy achieved was 99.87% (Table.2) using the Sigmoid Activation function.

Table 1. Achieved Accuracy for digitally created dataset with different parameters of the CNN

Activation Function	Filter Maps	Learning Rate	Batch Size	Accuracy Achieved
Rectified Linear Unit	30	0.1	30	99.79
Rectified Linear Unit	30	0.1	20	98.23
Sigmoid Activation Function	30	0.08	30	99.79
Tanh Activation Function	30	0.08	30	99.79

Table 2. Achieved Accuracy for hand drawn dataset with different parameters of the CNN

Activation Function	Filter Maps	Learning Rate	Batch Size	Accuracy Achieved
Rectified Linear Unit	30	0.16	30	98.23
Tanh Activation Function	30	0.16	30	97.78
Sigmoid Activation Function	30	0.16	20	99.87

5 K- Nearest Neighbour approach

The k-Nearest Neighbours algorithm compares images directly to observe the similarities between them. For a particular test element, it compares all the training elements with the test element and finds k different elements which are most similar to the test element. The similarity between two images can be observed by the value of the distance function.
The training element that gives the least value to the distance function on being compared with the test element is its nearest neighbour. K such neighbours are found and all their labels are compared. The label which occurs with the highest probability is the label that gets allotted to the test element.

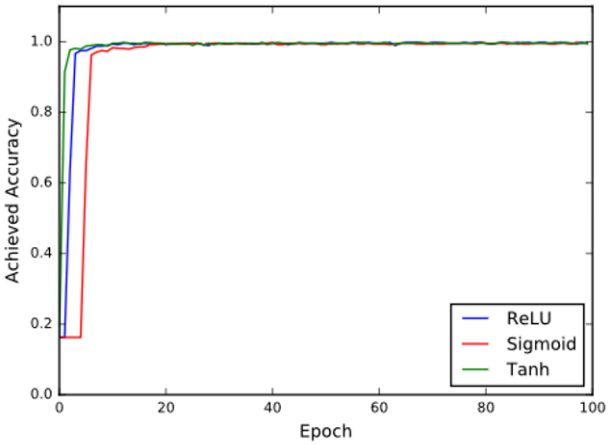


Fig. 6. Shows the learning curve for different activation functions used.

The advantage of using the nearest neighbour algorithm is its fairly fast build time which would only require indexing of the dataset. However, the computational time saved by the model during training would be nullified by the amount of time that would be required during the testing. Convolutional neural networks on the other hand, take a large amount of time to train, but once trained are easier to use.
The importance of taking into account k neighbours instead of just one is to ensure that outliers are not used to classify data and instead a more smoothened classification system is achieved. Very often the algorithm encounters cases where the labels found for two different neighbours have equal amount of occurrences. In those cases, no label is allotted to the test element. The distance function most commonly used is the Manhat-

Rows	Columns	Label
2	1	A
3	1	B
2	2	C
3	2	D
2	3	E
3	3	F

Fig. 7. Labels for the 6 classes.

tan function. Here, x_i is the pixel value of the pixel number i in the first image, y_i is the pixel value of the pixel number i in the second image. A sum of all the differences

($\sum_{i=1}^{k} |x_i - y_i|$) gives us the value of the distance which is used to liken the test image to its nearest neighbour.

Another distance function that is used very often is the Euclidean function which squares the value of the differences ($\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$) between pixel values before adding them up. One could easily prove how this function would have more affinity to many normal sized differences as compared to one big difference which would extrapolate the value of the distance function too far individually.

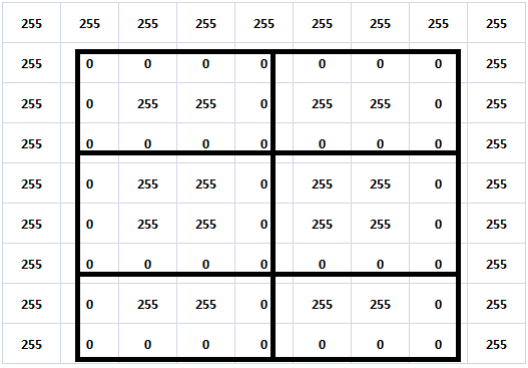


Fig. 8. Example of a table represented in pixel values. 0 being black and 255 white.

5.1 Results with knn

Table 3. Achieved Accuracy for digitally drawn tables

Distance Function	Manhattan	Euclidian
Accuracy	98.40	97.80

The accuracy of prediction for the Manhattan function is better than accuracy with the Euclidian function. The difference in the accuracies can be put down to the type of images being fed to the knn classifier. Since the images being used have lesser variation among them, only 6 different tables being used, the straightforward Manhattan distance function has a greater accuracy.

Table 4. Achieved Accuracy for Hand drawn tables

Distance Function	Manhattan	Euclidian
Accuracy	95.20	94.40

6 Conclusion

The average accuracy achieved for digitally drawn tables, by the neural network system is 99.3% and the average accuracy achieved by the knn system is 98.1%, and it is 98.6% and 94.8% respectively for hand drawn tables. It is safe to assume that the accuracies achieved are high enough for these algorithms to be used in development of software that can read not only the tables but also the data in them by using the pre-existing text-recognition tools. Further work can be done towards applying these techniques on a dataset that represents more real world variations.

References

1. Beyer, K., Goldstein, J., Ramakrishnan, R., Shaft, U.: When is nearest neighbor meaningful? In: International conference on database theory. pp. 217–235. Springer (1999)
2. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE transactions on information theory 13(1), 21–27 (1967)
3. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE transactions on information theory 13(1), 21–27 (1967)
4. Dudani, S.A.: The distance-weighted k-nearest-neighbor rule. IEEE Transactions on Systems, Man, and Cybernetics (4), 325–327 (1976)
5. Garcia, V., Debreuve, E., Barlaud, M.: Fast k nearest neighbor search using gpu. In: Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on. pp. 1–6. IEEE (2008)
6. Ghosh, A.K.: On optimum choice of k in nearest neighbor classification. Computational Statistics & Data Analysis 50(11), 3113–3123 (2006)
7. Le Cun, B.B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Handwritten digit recognition with a back-propagation network. In: Advances in neural information processing systems. Citeseer (1990)
8. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation 1(4), 541–551 (1989)
9. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11), 2278–2324 (1998)
10. LeCun, Y., Kavukcuoglu, K., Farabet, C., et al.: Convolutional networks and applications in vision. In: ISCAS. pp. 253–256 (2010)
11. Lee, Y.: Handwritten digit recognition using k nearest-neighbor, radial-basis function, and backpropagation neural networks. Neural computation 3(3), 440–449 (1991)
12. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
13. Smith, S.J., Bourgoin, M.O., Sims, K., Voorhees, H.L.: Handwritten character classification using nearest neighbor in large databases. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(9), 915–919 (1994)
14. Song, Z., Roussopoulos, N.: K-nearest neighbor search for moving query point. In: International Symposium on Spatial and Temporal Databases. pp. 79–96. Springer (2001)
15. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research 15(1), 1929–1958 (2014)

Development of Hands-free Speech Enhancement System for Both EL-users and Esophageal Speech Users

Yuta Matsunaga[1], Kenji Matsui[1], Yoshihisa Nakatoh[2], Yumiko O. Kato[3]

[1] Osaka Institute of Technology, 5-16-1, Omiya, Asahi-ku, Osaka, 535-8585, Japan
e1x13073@st.oit.ac.jp, kenji.matsui@oit.ac.jp
[2] Kyushu Institute of Technology, 1-1, Sensuichou, Tobata-ku, Kitakyushu, 804-8550, Japan
nakatoh@ecs.kyutech.ac.jp
[3] St. Marianna University School of Medicine, 2-16-1, Sugao, Miyamae-ku, Kawasaki, 216-8511, Japan
yumiko_o_kato@cwk.zaq.ne.jp

Abstract.
A hands-free speech enhancement system for laryngectomies is proposed to improve the usability and the speech quality. The system has a small and right weight transducer, photo-reflector based lip movement level sensors, and a microphone. The sensor outputs are used for generating the transducer turn on/off signal. A simple performance test was conducted if the system can determine the utterance segments accurately, and the result showed good utterance detection performance. Also, we tested with five more subjects and confirmed that the proposed method was well performed even under the speaker-independent condition.

Keywords: laryngectomies, electrolarynx, hands-free, esophageal, VOX

1 Introduction

People who have had laryngectomies have several options for restoration of speech, but currently available devices are not satisfactory.

The electrolarynx (EL), typically a hand-held device which introduces a source vibration into the vocal tract by vibrating the external walls, has been used for decades by laryngectomees for speech communication. It is easy to master with relatively short term practice period regardless of the post-operative changes in the neck. However, it has a couple of disadvantages. Firstly, it does not produce airflow, so the intelligibility of consonants is diminished and the speech is a very mechanical tone that does not sound natural. Secondly, one must use their hand to control the EL all the time, and it is far from normal appearance. Alternatively, esophageal speech does not require any special equipment, but requires speakers to insufflate, or inject air into the esophagus. It takes long time to master the speech, especially, elderly laryngectomees face difficulty in mastering the speech or keep using esophageal speech because of the waning strength. For that reason, some people who use esophageal speech also use the electrolarynx.

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5_41

The number of Tracheo-esophageal speech (TE) users has been increasing. It requires shunting between the trachea and the esophagus, when producing TE speech.

Both esophageal speech and tracheo-esophageal speech are characterized by low average pitch frequency, large cycle-to-cycle perturbations in pitch frequencies, and low average intensity. Often, portable loudspeakers are used in noisy environment.

EL on/off control is one of the important mechanisms for EL users to be able to generate naturally sounding speech. Most of the commercially available EL devices are using a single push button to produce F0-contours [1, 2, 3, 5]. There are also some F0-contour generation studies using additional control mechanism[4].

An EL system that has a hands-free user interface could be useful for enhancing communication by laryngectomees. Also, the appearance can be almost normal because users do not need to hold the transducer by hand against the neck. In our previous studies, it has been shown that motion sensor based EL controlling methods with forearm or finger motion are possible alternative methods. However, those methods are not completely hands-free and still requires slight hand movements.

The present study aims to develop highly practical speech enhancement system which can control EL device by utilizing lip motion information, so that the system can be completely hands-free. Also the system needs to be able to generate natural and clear utterance with no strange appearance.

2 User profile

The questionnaires were used to understand implicit user needs with 121 laryngectomees (87% male, 13% female), including 65% esophageal talkers, 12% EL users, 7% both, and 21% used writing messages to communicate.

We extracted primary needs of laryngectomees from the result as shown in Fig. 1.

Based on those survey results, the present study was conducted to meet the essential user needs.

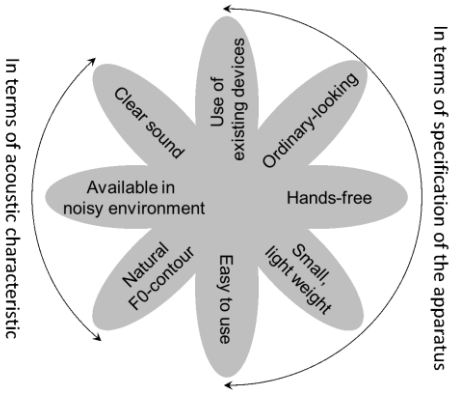


Fig. 1. The primary needs of laryngectomees

3 Speech enhancement system

Fig 2 shows a concept image of the proposed system.

A small and right weight EL transducer, which generates sound source, is put on user's neck. That device is controlled by the head-worn photo-reflector array. Users can control power on/off by moving lips.

A microphone is also put on together with the photo-reflector. It picks up the utterance generated by the EL transducer or esophageal speech. The picked up speech signal is sent to the wireless loudspeaker. Amplifying the speech output is necessary in case of esophageal speech, however, it is useful for EL users because the transducer output level can be lowered compare with usual EL devices. The sensor output can be used for controlling loudspeaker on/off function because the sensors can accurately detect the onset of speech. Therefore, the proposed system is also useful for people who use esophageal speech and tracheo-esophageal speech with loudspeaker system.

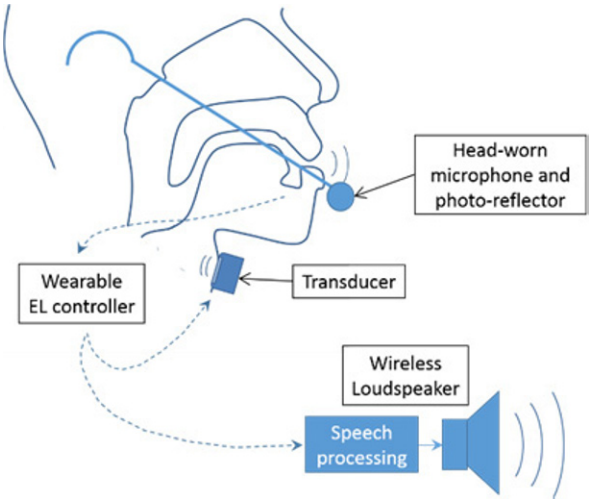


Fig. 2. A concept image of the proposed system

4 Implementation of hands-free user interface

A block diagram of photo-reflector based hands-free user interface is shown in Fig 3. We introduced the small hardware in order to meet the user requirements, i.e. small, comfortable weight, and low cost. The control algorithm were implemented on a small MPU board with VOX circuit and EL driver. To be able to stabilize the lip movement detection, four photo-reflectors are used and the outputs are added up. Fig 4 shows the head-worn photo-reflectors and ECM. As you can see in the Fig 5, the sensors were placed very close to the lips.

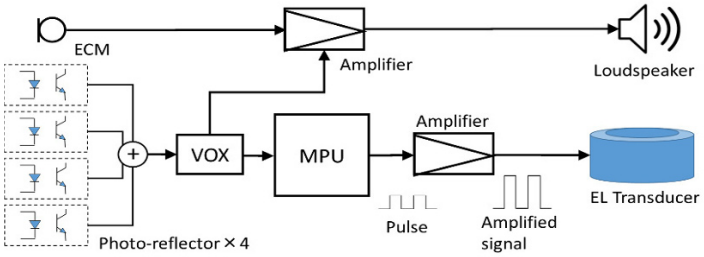


Fig. 3. A block diagram of hands-free user interface

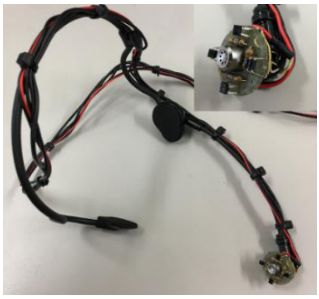


Fig. 4. Prototype of photo sensors

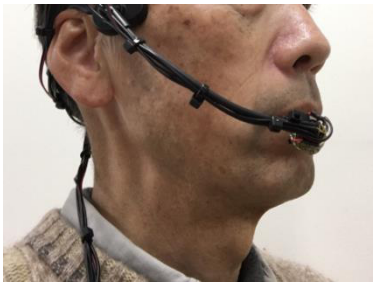


Fig. 5. Testing the photo sensors

4.1 Prototyping EL transducer

A small and thin transducer has also been prepared to generate voice source and it is placed to the so-called "sweet spot" on the neck with a neck-bandage. Fig 6 shows the EL transducer prototype. The diaphragm materials were tested and selected to make the speech output intelligible. Fig 7 shows the current transducer output frequency response.

Fig. 6. EL transducer prototype

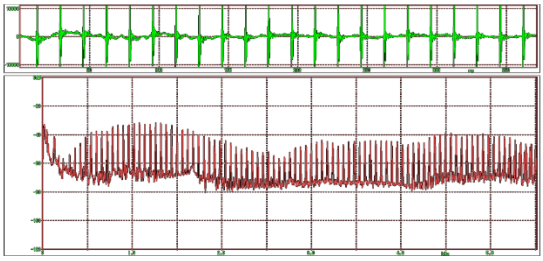


Fig. 4. EL transducer waveform and freq. response (500Hz/div, 20db/div)

4.2 Evaluation of VOX signal generation

Using the photo-reflector sensors, we tested VOX signal generation using one of the photo-reflector output. Fig 8 shows both speech signal and the VOX signal generated by the sensor 4.

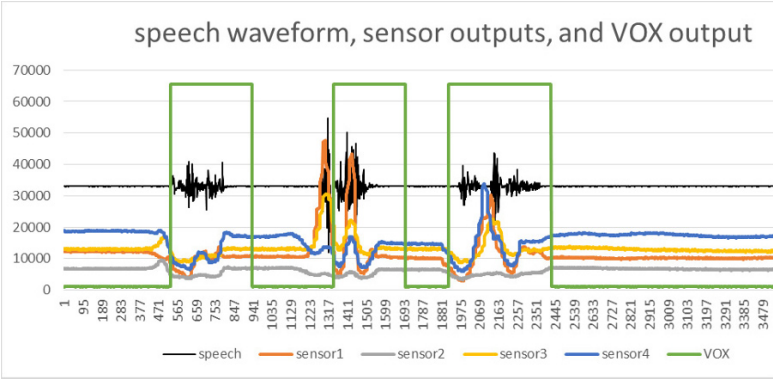


Fig. 5 VOX generation using sensor 4

One normal male speaker put the head set and uttered three phrases. The utterance is “mi na sa nn, ko re wa, te su to de su”. As you can see, four photo-reflector outputs show the utterance segments very clearly. The VOX circuit is able to detect the each utterance except beginning of the second utterance, i.e. “ko” in the “ko re wa”. That is because the VOX circuit is responding to falling edge of the sensor 4 output. To generate more precise VOX signal, $V(n)$ was used Eq. (1),

$$V(n) = \frac{V_{max}}{2} - \alpha \left| \sum_{i=1}^{4} \left(S_i(n) - S_i(n-m) \right) \right| \tag{1}$$

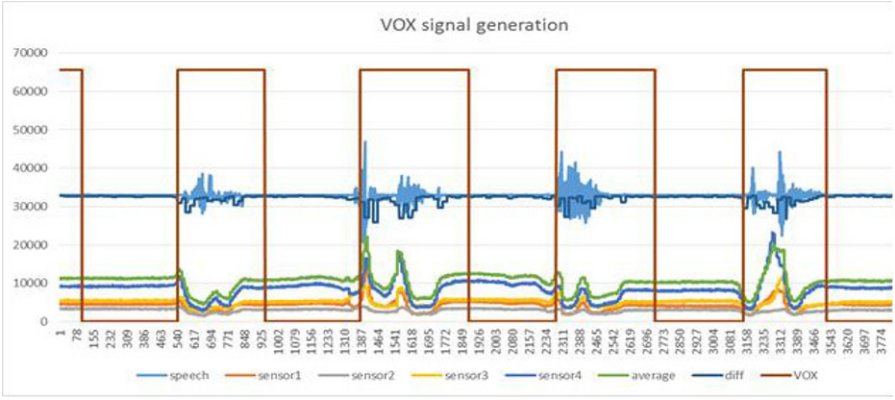


Fig. 6. VOX signal generation using V(n) (diff)

Where $Si(n)$ (i=1~4) are the sensor outputs, and n, m are the discrete time indices. V_{max} is maximum voltage of MPU output. α is a scaling factor. Fig 9 shows the generated VOX signal using $V(n)$.

The sampling rate is 2kHz, the delay m is 20 samples. The utterance is "mi na sa nn, ko nn ni chi wa, ko re wa, te su to de su". (The VOX circuit adds 0.2sec more at the end.)

In this case, $V(n)$ shows the phrase segments precisely.

4.3 Evaluation of VOX signal generation

Using the same sensor system, we asked five more subjects to see if the method is speaker-independent and robust enough. Each subject was asked to speak two phrases, ten times. The result showed quite stable performance with no detection error. We also found that the position of the sensors is very important to stabilize the performance. Since reliable EL ON/OFF control is very important for users to talk comfortably, EL ON/OFF control using the VOX signal seems promising.

4.4 F0 contour Generation

Based on the preliminary experimental results, the proposed photo-reflector-based EL user interface seems applicable even for the F0 contour generation. Previously, we developed the phrase component generation mechanism using motion sensor based UI [9, 10, 12]. To be able to stabilize the pitch contour generation, Fujisaki's model [11] was applied as a template. The phrase component of Fujisaki's model was used to generate the F0 template $F0(t)$.

$$\ln F0(t) = \ln F\min + Ap \cdot Gp(t) \qquad (1)$$

where

$$Gp(t) = \alpha^2 t \exp(-\alpha t) \qquad (2)$$

The symbols in equations (1) and (2) indicate: Fmin is the minimum value of speaker's F0, Ap is the magnitude of phrase command, and α is natural angular frequency of the phrase control mechanism. Fig 10 shows an example of the generated F0 contour. In this experiment, the rising edge of the VOX signal was used to turn on the F0 template based pitch signal. The utterance is "mi na sa nn, ko nn ni chi wa ko re wa te su to de su" without breath. Fig. 11 is the observed natural pitch contor.

In this study, those values are; Fmin = 80Hz, α=1.5, and Ap=0.75. The calculated F0 template data is stored in the controller software. The template duration is not controlled yet.

Currently, we are exploring machine learning tools to be able to extract pitch contour related information and accent related information from the photo-reflector sensor outputs. We are expecting to make reasonably accurate prosody estimation from those machine learning tool outputs.

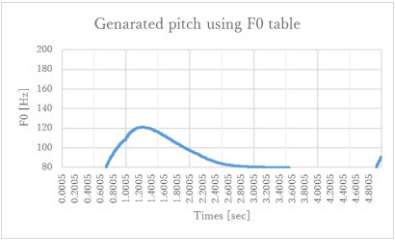


Fig. 10. Generated F0 contour

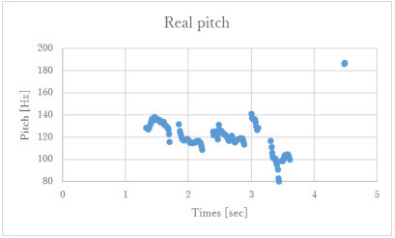


Fig. 11. Observed F0 contour

5 Discussion

Based on the preliminary experimental results, the proposed photo-reflector-based EL user interface seems applicable with small amount of additional hardware. It is also right-weight and less expensive. And more importantly, people can use EL device without using their hands. To be able to develop practical system, we need to test our proposed method with the total speech enhancement system. Also, we are very interested in generating naturally sounding F0 contour using the $V(n)$ [6, 7, 8]. Previously, we developed the phrase component generation mechanism using motion sensor based UI[9, 10, 12]. By modifying the previous system and using V(n) signal, we will investigate the possible way of generating the phrase component. Furthermore, if we will be able to control the prosody by lip movements, people might be able to learn how to make their prosody more natural. As for our next step, the above direction will be investigated.

6 Conclusions

A hands-free speech enhancement system for laryngectomees based on the photo-reflector was proposed. The total speech enhancement system using the specially designed EL and the photo-sensors were also proposed. The VOX function with the photo-reflector output was tested and found that lip movement signal from the proposed device has a potential to make the EL signal accurately without doing any hand operation. As our next step, we would like to focus on generating natural F0 pattern using the photo-reflector based UI.

Acknowledgment

This work was supported by JSPS KAKENHI Grant-in-Aid for Scientific Research(C) Grant Number 15K01487

References

1. DENSEI Communication Inc., YOURTONE (DELEY-2), http://www.dencom.co.jp/product/yourtone/yt2.html
2. SECOM Company Ltd., Electrolarynx “MY VOICE”, http://www.secom.co.jp/personal/medical/myvoice.html
3. Griffin Laboratories, TruTone Electrolarynx, https://www.griffinlab.com/Products/TruTone-Electrolarynx.html
4. Kikuchi,Y., Kasuya, H., "Development and evaluation of pitch adjustable electrolarynx", In SP-2004, 761-764, 2004
5. Takahashi, H., Nakao, M., Ohkusa, T., Hatamura,Y., Kikuchi, Y., Kaga, K., “Pitch control with finger pressure for electrolaryngial or intra-mouth vibrating speech”, Jp. J. Logopedics and Phoniatrics, 42(1),1-8, 2001
6. Nakamura, K., Toda, T., Saruwatari, H., Shikano, K., “The use of air-pressure sensor in electrolaryngeal speech enhancement”, INTERSPEECH, 1628-1631, Makuhari, Japan, Sept 26-30, 2010.
7. Saikachi, Y., “Development and Perceptual Evaluation of Amplitude-Based F0 Control in Electrolarynx Speech”, Journal of Speech, Language, and Hearing Research Vol.52, 1360-1369, October, 2009
8. Fuchs, A.K., Hagmüller, M. “Learning an Artificial F0-Contour for ALT Speech”, INTERSPEECH, Portland, Oregon, Sept. 9-13, 2012.
9. Kimura, K., Matsui, K., Nakatoh, Y., Kato, Y. O., Preliminary study of Hands-free speech enhancement system, autumn meeting, ASJ 2015, 401-402, September, 2015.
10. Kimura, K., Matsui, K., Nakatoh, Y., Kato, Y. O., Prosody control study of a hands-free speech enhancement system, autumn meeting, ASJ 2013, 395-396, September, 2013.
11. Fujisaki, H., Analysis of voice fundamental frequency contours for declarative sentences of Japanese, J. Acoust. Soc. Jpn, 233-242, 1984.
12. Kimura, K., Matsui, K., Nakatoh, Y., Kato, Y. O., Development of Wearable Speech Enhancement System for Laryngectomees, NCSP2016, 339-342, March, 2016.

Author Index

© Springer International Publishing AG 2018

S. Omatu et al. (eds.), *Distributed Computing and Artificial Intelligence, 14th International Conference*, Advances in Intelligent Systems and Computing 620, DOI 10.1007/978-3-319-62410-5